

UNCLASSIFIED
AD 413620

DEFENSE DOCUMENTATION CENTER

FOR

SCIENTIFIC AND TECHNICAL INFORMATION

CAMERON STATION, ALEXANDRIA, VIRGINIA



UNCLASSIFIED

**Best
Available
Copy**

NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

REPORT NO. 3

30 APRIL 1963

41 3620

CATALOGED BY DDC
AS AD No. _____

41 3620

NO. OTS

RESEARCH IN INFORMATION RETRIEVAL

Third Quarterly Report

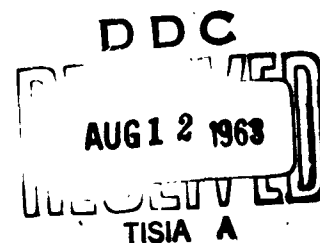
1 January 1963 - 31 March 1963

Contract No. DA 36-039-SC-90787

File No. 1160-PM-62-93-93(6509)

Technical Report P-AA-TR-(0044)

U. S. Army Electronics Research and Development Laboratory
Fort Monmouth, New Jersey



ITT

International Electric Corporation

Route 17 and Garden State Parkway, Paramus, New Jersey

A SUBSIDIARY OF INTERNATIONAL TELEPHONE AND TELEGRAPH CORPORATION

ASTIA AVAILABILITY NOTICE

**Qualified requestors may obtain copies
of this report from ASTIA.**

ASTIA release to OTS not authorized.

RESEARCH IN INFORMATION RETRIEVAL

Third Quarterly Report
1 January 1963 - 31 March 1963

An investigation
of the techniques and concepts of information retrieval

Contract No. DA 36-039-SC-90787

File No. 1160-PM-62-93-93(6509)

Signal Corps Technical Requirement
SCL-4218 12 January 1960

Technical Report P-AA-TR-(0044)

Jacques Harlow, Principal Investigator

prepared by
Alfred Trachtenberg
Quentin A. Darmstadt
George Greenberg
Alexander Szejman

1160-PM-62-93-93(6509)
TISIA A

TABLE OF CONTENTS

<u>Section</u>	<u>Title</u>	<u>Page</u>
	LIST OF ILLUSTRATIONS	iii
	LIST OF TABLES	iii
1	<u>PURPOSE</u>	1
1.1	Scope	1
1.2	Objectives	1
1.3	Project Tasks	1
1.3.1	Input Capability	2
1.3.2	Query Capabilities	7
1.3.3	Processing Techniques	10
1.3.4	Information Retrieval System Theory and Integration: Integration Capabilities	12
1.3.5	Summary	13
2	<u>ABSTRACT</u>	17
3	<u>PUBLICATIONS, REPORTS, AND CONFERENCES</u>	19
3.1	Technical Notes	19
3.2	Reports	19
3.3	Conferences	20
4	<u>FACTUAL DATA</u>	21
4.1	Organization	21
4.2	Input Capabilities	21
4.2.1	Information Theoretic Methods of Document Categorization Using Word Frequency Information	21
4.2.2	Non-Boolean Retrieval Processes	32
4.3	Query Capabilities	53
4.3.1	An Approach to a Criterion for Automatically Generated Extracts	53
4.3.2	The Problem of Redundancy in Information Retrieval Systems	57
4.4	Processing Capabilities	61

TABLE OF CONTENTS (Continued)

<u>Section</u>	<u>Title</u>	<u>Page</u>
	4.4.1 Comparative Analysis of Some File Organizations	61
	4.5 Integrative Capabilities	98
	4.5.1 General Theoretical Considerations with Special Reference to the Relationship Between Frequency and Indexing	98
	4.6 References	107
5	<u>CONCLUSIONS</u>	109
6	<u>PLANS FOR THE NEXT QUARTER</u>	111
7	<u>IDENTIFICATION OF PERSONNEL</u>	113
	7.1 Personnel Assignments	113
	7.2 Background of Personnel	113
	<u>DISTRIBUTION LIST</u>	115

LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Title</u>	<u>Page</u>
1	Average Number of Headings and Items Examined in a Search of Differently Organized Files	84
2	Number of Levels Required to Store N Items in a Regular Tree	87
3	Standard Deviation From Average Number of Headings and Items Examined in a Search	92
4	Cumulative Probability Distributions for a Search of Differently Organized Files	94

LIST OF TABLES

<u>Table</u>	<u>Title</u>	<u>Page</u>
1	Probabilistic Assignment	34
2	Summary of File Organizations	74

1. PURPOSE

1.1 SCOPE

This report discusses the work performed for the U. S. Army Signal Electronics Research and Development Laboratory (USAERDL) under Contract No. DA 36-039-SC-90787 during the period from 1 January 1963 to 31 March 1963.

1.2 OBJECTIVES

The objective of this project is to investigate the techniques and concepts of information retrieval and to formulate and develop a general theory of information retrieval. The formalisation of this theory is oriented to the automation of large-capacity information storage and retrieval systems. This theoretical framework will be the basis for the use of general purpose stored-program digital computer systems to perform the storage and retrieval functions.

1.3 PROJECT TASKS

The task structure described in this section is based upon the information retrieval model specified in the First Quarterly Report to USAERDL, the framework elaborated for it in the Second Quarterly Report, and subsequent discussions with USAERDL project personnel. This structure is intended as an organizational guide for continuing investigations. It is not intended to exclude constructive effort in task areas that may not have been foreseen, nor is it likely that all the tasks and subtasks specified will receive equally intensive treatment.

The goal of this project is a theory or a model of a fully automated information content storage and retrieval systems. The task structure

deals with four areas of procedural capability that must be developed if this goal is to be achieved:

- (a) Input capabilities.
- (b) Query capabilities.
- (c) Processing capabilities.
- (d) Information retrieval system theory and integration (integrative capabilities).

The first three areas are roughly analogous to the D, E, and P transforms of the basic information retrieval model. The last area is a supra-ordinate category that indirectly involves the other three. Each of these areas will be briefly considered as tasks, salient subtasks will be described, and the interrelationships between various tasks and subtasks will be pointed out.

1.3.1 Input Capability - In the ultimate system written, printed, or oral material in natural language should be accepted as input for automatic processing and analysis at the morphological, syntactic, semantic, logical, and factual levels. As a consequence of such input processing, all explicit and implicit or factual reference of the input material should be appropriately displayed or elucidated for further processing in response to queries.

In large measure, most of these potential capabilities are outside the scope of this project. Visual or auditory pattern recognition devices for reading or listening to natural language are an ancillary problem that may be left for separate development. Linguistic analysis has been eliminated as a primary focus of the project, and an attempt to

achieve mechanical understanding of text is thus also beyond the scope of this research activity.

Whether read-in and linguistic analysis are completely automated or not, a central problem in the transformation of information inputs into forms useable in storage and retrieval is the classifying, categorising, or indexing process. Such a classification stage is essential regardless of the degree of sophistication or automation in an information storage and retrieval system.

Capabilities in this area are currently quite limited. To date operational classificatory schemes tend to be intuitively formulated and manually implemented. Furthermore, there are no systematic procedures for improving the precision of categories to assure that the denotations used by the system properly correspond to the denotations understood or desired by the user. Accordingly there are three major subtasks:

- (a) The development of explicit procedures for establishing useful category groupings and boundaries.
- (b) The development of procedures for automatically assigning items to classificatory categories.
- (c) The development of methods for improving the precision of category denotation between the system and the user.

Before considering each subtask in further detail, it should be noted that they need not be completely independent. Ultimately, these capabilities cannot be fully developed without reference to other system capabilities--i.e., query and processing. Furthermore, subtasks (a) and (b) may merge into a single theoretical and procedural statistical scheme for both selecting categories and assigning items to them. Such interdependence

in dynamic systems does not, however, preclude an essentially parallel attack on separate aspects of the capabilities problem.

1.3.1.1 Development of Explicit Procedures for Establishing Useful Category Groupings and Boundaries - There is already a mathematical literature on the problem of category formation based upon measures of relevance between the units to be grouped. This literature will not be reviewed here, such a review being a preliminary phase of work on each of the tasks and subtasks. Some of the relevant work involves factor analysis, latent class analysis, and the theory of clumps.

There are two kinds of applications for such explicit procedures in establishing categories:

- (a) For grouping or indexing documents--i.e., items received by the information system--into larger categories. This application is essentially the same function that intuitive library classification schemes currently serve.
- (b) For finding salient boundaries within documents or items to analyze them into smaller useable parts. As the number of parts increases and the articulation of their interrelationships increase in sophistication, the goal of input analysis is approached in evolutionary stages.

1.3.1.2 Development of Procedures for Automatically Assigning Items to Classificatory Categories - The purpose of this subtask is twofold: first, to eliminate subjectivity in the classification of library

items and thus to increase precision; second, to alleviate the time and cost required for manual classification. It is irrelevant to these purposes whether the classificatory scheme is systematically developed as membership in exclusive categories or whether a traditional scheme is to be implemented automatically. There may, however, be differences in the complexity and difficulty of the automatic classification problem based upon the type of classificatory schemes used. Specifically, the question of independence among classificatory categories and type of class membership may affect the nature of the automatic classificatory procedures.

Two kinds of problems can be distinguished as follows:

- (a) Membership in exclusive categories. This situation exists when categories are exclusive. An item can be assigned to only one category and not assigned to the remainder. Clue word schemes developed to date, including the approach reported in the Second Quarterly Report, are essentially limited to such classification. This type of classification exists in traditional hierarchical schemes such as the Dewey or Library of Congress systems. Such hierarchies, if well conceived, have the advantage that cruder discriminatory or predictive techniques can be applied to higher level distinctions until more precise methods are available for dealing with lower level distinctions.
- (b) Degree of inclusion and/or nonexclusive categories. This problem is a more general case in which an item may be not only assigned to a given category, but also assigned to some degree as relevant

to a category--and/or assigned to more than one category simultaneously. The systems included under exclusive categories are a special case of nonexclusive categories, and the general case will require more sophisticated treatment. While operational systems do not yet extensively use category assignment by degree of relevance, newer Uniterm or coordinate indices already use multiple category assignment per document. As increasingly articulated category assignment becomes possible automatically, the ultimate goal of the project is approached.

1.3.1.3 Development of Methods for Improving the Precision of Category Denotation Between the System and the User - Assuming that categories and item assignment have somehow been arrived at, whether intuitively and manually or explicitly and automatically, the system cannot function optimally unless category denotation agrees with usage. Since it is unlikely either that the system's denotations will agree perfectly with those of the average user or that the denotations of the users will agree perfectly, there are two kinds of problems that can currently be isolated within this subtask:

- (a) Corrective procedures. These procedures refer to the application of user feedback, along with assumed invariances between the user and system denotations, to adjust the assigned item content in the system's categories. A fuller account of an approach to this problem is included in the Second Quarterly Report.

(b) Non-Boolean retrieval. This function refers to the problem of using criteria for averaging or optimizing category membership under conditions of user disagreement. It is not generally the case that an optimization criterion for category membership--e.g., 50 percent user agreement on an item places it within the category--will be fulfilled for Boolean functions of individually optimized categories. That is, the union of two 50-percent agreement categories may not contain only those documents on which there was 50-percent agreement that they belong in both categories. Hence, non-Boolean retrieval functions are needed to resolve this problem.

1.3.2 Query Capabilities - Many of the general considerations regarding pattern recognition, linguistic analysis, and problem interrelations discussed under input capabilities are also relevant as functional aspects of queries. The situation is so similar, however, that a repetition of this discussion in the query capability context is unnecessary. As in the case of input capabilities, the query problem will be attacked from the viewpoint of relaxing the limitations of current information storage and retrieval systems.

In most operational systems the possible query is essentially:

"What documents in the system contain information of the following kind _____?"

There are at least three limitations on this form of query that require resolution before more sophisticated information storage and retrieval systems are possible:

- (a) Limitation to documents.
- (b) Limitation to unrestricted retrieval of all items.
- (c) Limitation on description of type of information desired.

Each of these limitations will be considered as subtasks.

1.3.2.1 Limitation to Documents - The query capability should be extended so that a system may respond with appropriate portions of documents rather than documents as a whole. The input capability described under explicit techniques for salient boundaries is essential for satisfying this query capability. Another approach might involve the extension of work already done in automatic abstracting or extracting, which selects salient information from documents rather than merely salient portions of documents.

Such a capability cannot be provided in a vacuum. Input capabilities must provide indices to document parts as well as isolate them. Processing capabilities must provide means of associating such document parts with the query. These considerations apply equally to the remaining subtasks considered under query capabilities.

1.3.2.2 Limitation to Unrestricted Retrieval of All Items - The purpose of this subtask is essentially the same as that for the preceding one--viz., to reduce necessary search activity on the part of the user by performing it within the system. Only in specialized scholarly situations does the user need all documents that are potentially relevant to his query. There are two problem areas suggested for this subtask:

(a) Elimination of (low quality) redundancy. In many fields there is a proliferation of documents covering the same topics. Many of these documents may also be low in quality. It is desirable to increase the sophistication of indexing, an input capability essentially, so that the contents of an item, even if it is only part of a document, are described or classified not only according to what topics they are relevant but also according to the degree of uniqueness the topics are dealt with. It seems that such indexing could not be readily achieved using purely statistical means and this capability may be one of the most difficult to automate.

(b) Specification of scope. In addition to weeding out redundant or low quality materials, it would be desirable to be able to restrict the scope of retrieval on a given query according to the needs of the user. This function obviously would involve considerations of relevance and its measurement as well as integration with the mode in which desired information is characterized. The latter requirement is also considered in the following subtask.

1.3.2.3 Limitation on Description of Type of Information Desired -

Different operational information systems impose different limitations of this type. A hierarchically organized index or query language may produce such unusual classifications of new material that a subsidiary index is necessary in order to use the primary index properly. Freer Uniterm systems are limited to Boolean functions of two-valued descriptors; the

descriptor is either present or absent. The use of role indicators and similar devices offer some possibility of improving the query. But the crux of the problem is to develop a query capability that allows a user to state his question precisely. This ability is essential to useful content retrieval.

It should be noted that the problem of designing an adequate query or descriptor language for the purposes of the user has an analog in the design of an adequate representation of this language for machine processing. The design of the query language must, therefore, take into consideration problems of machine representation and processing as well.

1.3.3 Processing Capabilities - Advances in information storage and retrieval depend upon improved processing algorithms. Advances in the other capabilities will influence the choice of processing techniques. It is, consequently, difficult to define relatively independent problems in advance. In the present state of development, the processing task can be subdivided into two major subtasks:

- (a) Associative techniques.
- (b) Organization and search.

1.3.3.1 Associative Techniques - In order to respond to queries with appropriately indexed documents, an information system must have techniques for associating the two. In simple systems queries and index categories are so limited in differentiation that the association problem may become trivial. As greater flexibility is introduced in the query language and as input capabilities are improved, supplying appropriate

information requires associative capabilities.

One aspect of this problem, the measurement of relevance, has already been considered in the First Quarterly Report. Such measures are relevant both to input and query capabilities as well as to associative processing in response to queries. Further work is required on the development of associative techniques using such measures of relevance.

1.3.3.2 Organization and Search - There is a sense in which file organization may differ from search theory and procedures. At a system design level, however, these considerations become inseparable. Thus, while file organization may be abstractly distinguished from the procedures used to search a file, in practice the theoretical work in one area depends upon extensive explicit or implicit assumptions about the other. Accordingly, organization and search are treated in a single subtask.

Both organization and search, however, can be conveniently divided into two aspects, logic and efficiency.

- (a) Logical aspects. The logical aspects refer to organization or search procedures based upon logical relations that are inherent in the subject matter and the system and are essential to performing the processing. Examples of logical organization are alphabetization, hierarchies, or matrices.
- (b) Efficiency. Superimposed on a given logical organization are considerations of efficiency. These problems are most influenced

by the relative activity of different portions of the system, the nature of the information in the system, and the physical nature of the system. Efficiency considerations lead to rearrangements within a given logical organization for performing a system task at a minimum cost.

1.3.4 Information Retrieval System Theory and Integration: Integration Capabilities - This task did not appear as a separate unit in the Second Quarterly Report. At that time it appeared that it could be handled under processing. That report noted that some of the tasks included under processing were of a supra-ordinate nature, referring to the capabilities of information systems as a whole rather than to specific input, query, or processing capabilities. After reviewing the framework in that report, it was decided to consider these factors as a separate task.

There are three subtasks in this area:

- (a) Measures and models of system value and efficiency.
- (b) Models and methods for system integration and optimization.
- (c) General theoretical considerations.

It is apparent from this simple enumeration that while such considerations must permeate work in the other three areas of capability, a separate treatment is warranted in a project aimed at the development of a general information system theory or design methodology. Each of the subtasks will now be briefly considered.

1.3.4.1 Measures and Models of System Value and Efficiency -

This subtask is addressed to the development of a capability to answer

such questions as:

Do we need a new information system?

If so, what is its value?

What kind of system would most efficiently serve our needs?

Value and efficiency do not refer merely to the cost and specifications of individual pieces of hardware. Such engineering problems must ultimately be resolved in the design of any given system, but detailed consideration of these factors is outside the scope of this project. Value and efficiency thus refer to theoretical measures and models of the necessity and adequacy of the system as a whole.

1.3.4.2 Models and Methods for System Integration and Optimization - This subtask will deal with the problem of integrating specific configurations. Work on this subtask is to some extent dependent upon value and efficiency models, but the focus is upon theoretical methods rather than specific engineering considerations.

1.3.4.3 General Theoretical Considerations - This subtask is included to allow for work on the development of ideas that may emerge on the nature of information storage and retrieval. It constitutes an admission that the task and subtask structure may not yet contain the germinal or organising principles for a general theory of information retrieval.

1.3.5 Summary - A task framework has been described in terms of areas of capability that require development in order to evolve fully automatic, factual content, information storage and retrieval systems.

An outline of the task framework follows.

- (a) Input capabilities.
 - (1) Explicit procedures for establishing useful category groupings and boundaries.
 - a. Larger groupings.
 - b. Internal boundaries.
 - (2) Procedures for automatically assigning items to classificatory categories.
 - a. Exclusive categories.
 - b. Non-exclusive categories.
 - (3) Methods for improving the precision of category denotation between system and user.
 - a. Corrective procedures.
 - b. Non-Boolean retrieval.
- (b) Query capabilities.
 - (1) Relax limitation to documents.
 - a. Portions of documents.
 - b. Abstracts or extracts.
 - (2) Restricted retrieval.
 - a. Elimination of redundancy.
 - b. Specification of scope.
 - (3) Relax limitations on description.
- (c) Processing capabilities.
 - (1) Associative techniques.
 - (2) Organisation and search.
 - a. Logic.
 - b. Efficiency.

(d) Integration capabilities.

- (1) Measures and models of system value and efficiency.
- (2) Models and methods for system integration and optimization.
- (3) General theoretical considerations.

An attempt to classify both work planned and accomplished, as well as literature reviews, will continue in terms of the task framework presented in this section. Such a process will allow the framework to be articulated or revised as it is tested in practice.

2. ABSTRACT

Work in each of the four areas of capability isolated in the project task structure has been performed in the past quarter. Under input capabilities an extension of last quarter's work on procedures for automatic assignment has been accomplished and the development of a probabilistic non-Boolean retrieval model has been initiated. Under query capabilities new approaches to the problems of limitation to documents, automatic extracting or abstracting, and restricted retrieval--elimination of redundancy--have been developed.

The work on non-Boolean retrieval is also relevant to the query capabilities subtasks concerned with the specification of scope and the relaxation of limitations on descriptions. Under processing capabilities there is no new progress to be reported on associative procedures, but extensive mathematical analysis has been initiated on the problem of file organization and search. Finally, under integrating capabilities some general theoretical considerations have evolved that should lead to measures and models of system value and optimization in item retrieval systems.

3. PUBLICATIONS, REPORTS, AND CONFERENCES

3.1 TECHNICAL NOTES

The following internal technical memoranda were issued during this reporting period:

- (a) IEC TECHNICAL NOTE, File No. P-AA-TN-(0050)-N, 18 February 1963;
Task Framework for Continuation of Information Retrieval Research, George Greenberg, Quentin A. Darmstadt, Alexander Szejman, and Alfred Trachtenberg.
- (b) IEC TECHNICAL NOTE, File No. P-AA-TN-(0051)-N, 25 February 1963;
Analysis of File Organizations for Information Retrieval, Quentin A. Darmstadt.
- (c) IEC TECHNICAL NOTE, File No. P-AA-TN-(0058)-N, 19 March 1963;
An Approach to a Criterion for Automatic Extracts, George Greenberg and Alexander Szejman.
- (d) IEC TECHNICAL NOTE, File No. P-AA-TN-(0064)-N, 25 March 1963;
Non-Boolean Retrieval Processes, Alexander Szejman.
- (e) IEC TECHNICAL NOTE, File No. P-AA-TN-(0069)-N, 25 March 1963;
The Problem of Redundancy in the Information Retrieval Systems, Alexander Szejman.
- (f) IEC TECHNICAL NOTE, File No. P-AA-TN-(0070)-N, 25 March 1963;
Information Theoretical Methods of Document Categorisation Using Word Frequency Information, Alfred Trachtenberg.

These technical notes are dated at the time of their completion; these dates do not necessarily correspond to the date of publication.

3.2 REPORTS

The following reports were issued during this reporting period:

- (a) RESEARCH IN INFORMATION RETRIEVAL: Second Quarterly Report, 1 October 1962 - 31 December 1962, Technical Report P-AA-TR-(0031), (Manuscript Version), 31 January 1963.

- (b) MONTHLY LETTER REPORT NO. 5, 1 January 1963 - 31 January 1963, File No. P-AA-TR-(0032), 31 January 1963; Research in Information Retrieval, Alfred Trachtenberg.
- (c) MONTHLY LETTER REPORT NO. 6, 1 February 1963 - 28 February 1963, File No. P-AA-TR-(0033), 28 February 1963; Research in Information Retrieval, Alfred Trachtenberg.

3.3 CONFERENCES

The following conferences were held between IEC personnel and the USAERDL:

- (a) 28 February 1963--Meeting at IEC. IEC personnel met with Mr. Anthony V. Campi, who had recently been assigned as Project Engineer. Several aspects of the Second Quarterly Report were discussed. Several minor corrections and elaborations were requested. A general emphasis on the importance of user requirements was indicated.

4. FACTUAL DATA

4.1 ORGANIZATION

This section is organized according to the four major areas of required capability isolated in the project task structure (see Section 1.3).

4.2 INPUT CAPABILITIES

Work performed under input capabilities includes both an extension of the last quarter's work on information theoretic methods of document categorization using word frequency information and the development of a scheme for non-Boolean retrieval. This work has proceeded on Sections (a)(2) and (a)(3) of the task framework. Work on (a)(2), however, is also relevant to (a)(1). Furthermore, the work on non-Boolean retrieval has implications that are more general than input capabilities.

4.2.1 Information Theoretic Methods of Document Categorization Using Word Frequency Information

4.2.1.1 Introduction - In the last quarterly report, some information theoretical methods of document classification were presented. These methods used word occurrences as clues to the classification of a document. The number of times a word occurred in a document was not considered at that time; only the fact of its occurrence in a document was used to predict document categories. Thus all the information provided by word frequency information was neglected. This extension considers how such information can be used to provide a better prediction of categories of documents.

It was assumed that initially a group of human experts would classify a number of documents into a given set of categories, and that this initially classified group was large enough accurately to reflect the statistics of the larger body of documents that would later be automatically classified. Thus the probabilities of categorization of the larger group of documents were approximated by the relative frequencies of categorization of the initial group of documents.

The criteria used for selecting a particular word to predict categories were:

- (a) That its occurrence in documents be strongly correlated with the appearance of those documents in a particular category, for the group of documents that would be initially classified.
- (b) That the word supply more information than the a priori distribution of documents in categories did; i.e., that the distribution of documents containing this word differ markedly from the distribution of all the documents.

These criteria were expressed mathematically by the expressions:

$$H_i = - \sum_j p_{ij} \log p_{ij} \quad \left\{ \begin{array}{l} i = 1, \dots, m \\ j = 1, \dots, k \end{array} \right\} \quad (4-1)$$

and

$$M_i = \sum_j p_{ij} \log \frac{p_{ij}}{p_j}$$

where p_j is the probability that a document falls into category C_j , and p_{ij} is the probability that a document containing word W_i falls into category C_j . Thus a good predictor would have a low H_i and a high M_i .

4.2.1.2 Extension of Concepts to Include Word Frequency Information - There are several ways in which word frequency information can be taken into account to determine good predictors of document categories. The first two methods use absolute values of word occurrence in a document,

while the third method uses relative word frequency in a document to obtain more information.

Let:

N = the total number of documents in the initial group.

N_i = the number of documents in which word W_i occurs.

$N_i(x)$ = the number of documents in which word W_i occurs x times.

n_j = the number of documents in category C_j .

n_{ij} = the number of documents in category C_j which have word W_i .

$n_{ij}(x)$ = the number of documents in category C_j which have word W_i x times.

Now:

$$\left. \begin{aligned} N_i &= \sum_x N_i(x) \\ n_{ij} &= \sum_x n_{ij}(x) \end{aligned} \right\} (4-2)$$

In addition to the probabilities p_{ij} and p_j , the following probabilities can be defined. Let:

p_i = the probability that a document contains word W_i .

$p_i(x)$ = the probability that a document contains word W_i x times.

$p_{ij}(x)$ = the probability that a document containing word W_i x times falls into category C_j .

$p(C_j, W_i)$ = the joint probability that a document is in category C_j and contains word W_i .

$p[C_j, W_i(x)]$ = the joint probability that a document is in category C_j and contains word W_i x times.

Then the probabilities can be approximated as follows:

$$\begin{aligned}
p_i &= \frac{N_i}{N} \\
p_j &= \frac{n_j}{N} \\
p_{ij} &= \frac{n_{ij}}{N_i} \\
p_i(x) &= \frac{N_i(x)}{N} \\
p_{ij}(x) &= \frac{n_{ij}(x)}{N_i(x)} \\
p(C_j, W_i) &= \frac{n_{ij}}{N} \\
p[C_j, W_i(x)] &= \frac{n_{ij}(x)}{N}
\end{aligned}
\tag{4-3}$$

Of course:

$$\begin{aligned}
p_i &= \sum_x p_i(x) \\
p(C_j, W_i) &= \sum_x p[C_j, W_i(x)]
\end{aligned}
\tag{4-4}$$

and $p_{ij}(x)$ is related to p_{ij} by the expression;

$$p_{ij} = \frac{\sum_x p_{ij}(x) N_i(x)}{\sum_x N_i(x)}
\tag{4-5}$$

- (a) Method 1 - The measures H_i and M_i can easily be generalized to include frequency information by considering word W_i occurring x and only x times in a document as a clue. Then, instead of using p_{ij} in H_i and M_i , a new probability $p_{ij}(x)$ can be used.

Two new measures, $H_1(x)$ and $M_1(x)$, can now be defined:

$$\left. \begin{aligned} H_1(x) &= - \sum_j p_{1j}(x) \log p_{1j}(x) \\ M_1(x) &= \sum_j p_{1j}(x) \log \frac{p_{1j}(x)}{p_j} \end{aligned} \right\} \quad (4-6)$$

With these measures, the effectiveness of word W_1 as a predictor, when it occurs x times in a document, can be evaluated. As before, $H_1(x)$ must be low and $M_1(x)$ must be high for a good predictor.

The average effectiveness of a word W_1 as a predictor can be measured by:

$$\left. \begin{aligned} \overline{H_1(x)} &= \langle H_1(x) \rangle_x \\ \overline{M_1(x)} &= \langle M_1(x) \rangle_x \end{aligned} \right\} \quad (4-7)$$

Then, on the basis of Equations 4-3 and 4-4, it follows that:

$$\overline{H_1(x)} = \frac{\sum_x p_1(x) H_1(x)}{\sum_x p_1(x)} \quad (4-8)$$

and;

$$\overline{H_1(x)} = - \frac{1}{p_1} \sum_x \sum_j p[C_{j,W_1}(x)] \log p_{1j}(x) \quad (4-9)$$

Similarly:

$$\overline{M_1(x)} = \frac{\sum_x p_1(x) M_1(x)}{\sum_x p_1(x)} \quad (4-10)$$

But:

$$M_1(x) + H_1(x) = - \sum_j p_{1j}(x) \log p_j \quad (4-11)$$

therefore;

$$\begin{aligned} \overline{\langle M_1(x) + H_1(x) \rangle_x} &= \overline{M_1(x)} + \overline{H_1(x)} \\ &= - \frac{1}{p_1} \sum_x \sum_j p[C_j, W_1(x)] \log p_j \\ &= - \frac{1}{p_1} \sum_j p(C_j, W_1) \log p_j \end{aligned} \quad (4-12)$$

and, by substituting Equation 4-3;

$$\overline{M_1(x)} + \overline{H_1(x)} = - \sum_j p_{1j} \log p_j \quad (4-13)$$

But:

$$M_1 + H_1 = - \sum_j p_{1j} \log p_j \quad (4-14)$$

therefore;

$$\overline{M_1(x)} + \overline{H_1(x)} = M_1 + H_1 \quad (4-15)$$

- (b) Method 2 - This method is similar to Method 1. Instead of considering that a word occurs exactly x times in a document, this method considers that a word occurs between x_a and x_b times in a document. In other words, word frequency information is grouped in intervals of frequency of occurrence, B_x . For example, the frequency intervals might be 1-5 times, 6-10 times, etc.

New probabilities must be introduced. Let:

$p_1(B_r)$ = the probability that a document contains word W_1 x times, where x is in interval B_r .

$p_{1j}(B_r)$ = the probability that a document containing word W_1 x times falls into category C_j , where x is in interval B_r .

$p[C_j, W_1(B_r)]$ = the joint probability that a document is in category C_j and contains word W_1 x times, where x is in interval B_r .

Now the probabilities can be expressed as:

$$\left. \begin{aligned} p_1(B_r) &= \sum_{x \in B_r} p_1(x) \\ p[C_j, W_1(B_r)] &= \sum_{x \in B_r} p[C_j, W_1(x)] \\ p_{1j}(B_r) &= \frac{\sum_{x \in B_r} p_{1j}(x) N_1(x)}{\sum_{x \in B_r} N_1(x)} \\ &= \frac{\sum_{x \in B_r} p[C_j, W_1(x)]}{\sum_{x \in B_r} p_1(x)} \end{aligned} \right\} \quad (4-16)$$

Then, following Method 1 and Equation 4-6, expressions may be written for $H_1(B_r)$ and $M_1(B_r)$.

$$\left. \begin{aligned} H_1(B_r) &= - \sum_j p_{1j}(B_r) \log p_{1j}(B_r) \\ M_1(B_r) &= \sum_j p_{1j}(B_r) \log \frac{p_{1j}(B_r)}{p_j} \end{aligned} \right\} \quad (4-17)$$

$H_1(B_r)$ should be low and $M_1(B_r)$ should be high for a good predictor.

Another set of functions that measure the effectiveness of word W_1 as a predictor, when W_1 occurs x times and x is in interval B_r , can be obtained by taking the average values of $H_1(x)$ and $M_1(x)$ over the interval B_r . The average effectiveness is measured by:

$$\left. \begin{aligned} \overline{H_1(x,r)} &= \langle H_1(x) \rangle_{x \in B_r} \\ \overline{M_1(x,r)} &= \langle M_1(x) \rangle_{x \in B_r} \end{aligned} \right\} \quad (4-18)$$

Then, by using Equation 4-11 as in Method 1:

$$\begin{aligned} \langle H_1(x) + M_1(x) \rangle_{x \in B_r} &= - \frac{\sum_{x \in B_r} p_1(x) p_{1j}(x) \log p_j}{\sum_{x \in B_r} p_1(x)} \\ &= - \frac{1}{p_1(B_r)} \sum_j p[C_j, W_1(B_r)] \log p_j \\ &= - \sum_j p_{1j}(B_r) \log p_j \end{aligned} \quad (4-19)$$

But:

$$H_1(B_r) + M_1(B_r) = - \sum_j p_{1j}(B_r) \log p_j \quad (4-20)$$

therefore;

$$\begin{aligned} H_1(B_r) + M_1(B_r) &= \langle H_1(x) + M_1(x) \rangle_{x \in B_r} \\ &= \overline{H_1(x,r)} + \overline{M_1(x,r)} \end{aligned} \quad (4-21)$$

If this quantity $[H_1(B_r) + M_1(B_r)]$ is averaged over all r , then by the proof outlined for Method 1:

$$\begin{aligned}
H_1 + M_1 &= \overline{H_1(x)} + \overline{M_1(x)} \\
&= \overline{H_1(B_r)} + \overline{M_1(B_r)} \\
&= \langle H_1(x,r) \rangle_r + \langle M_1(x,r) \rangle_r
\end{aligned} \tag{4-22}$$

Thus the sum of the averages of the two measures remains constant and is independent of the size of the intervals of frequency of occurrence.

- (c) Method 3 - This method considers the number of times a word appears in a document in relation to the total number of words in a document as a clue. Using this relative frequency information as clues should provide even better category prediction than word occurrence or simple word frequency information.

Let f be the relative frequency of a word in a document; the relative frequency is the ratio of the number of occurrences of the word in the document to the total number of words in the document. Let f_s be an interval of relative frequencies, where the interval is defined by the limits f_a and f_b . Then, $p_1(f_s)$ is simply the probability of word W_1 occurring in a document with a relative frequency in the interval f_s , and $p_{1j}(f_s)$ is the probability that a document falls in category C_j , given that the document contains word W_1 with a relative frequency within the interval f_s .

The probabilities $p_1(f_s)$ and $p_{1j}(f_s)$ are approximated by:

$$\left. \begin{aligned} p_i(f_s) &= \frac{N_i(f_s)}{N} \\ p_{ij}(f_s) &= \frac{n_{ij}(f_s)}{N_i(f_s)} \end{aligned} \right\} \quad (4-23)$$

where $N_i(f_s)$ is the number of documents containing word W_i with a relative frequency within the interval f_s , and $n_{ij}(f_s)$ is the number of documents in category C_j containing word W_i with a relative frequency within the interval f_s .

Following the previous analyses, expressions for $H_i(f_s)$ and $M_i(f_s)$ can be written:

$$\left. \begin{aligned} H_i(f_s) &= - \sum_j p_{ij}(f_s) \log p_{ij}(f_s) \\ M_i(f_s) &= \sum_j p_{ij}(f_s) \log \frac{p_{ij}(f_s)}{p_j} \end{aligned} \right\} \quad (4-24)$$

By analogy to the proofs developed for Methods 1 and 2, $\overline{M_i(f_s)} + \overline{H_i(f_s)}$ can be calculated where:

$$\left. \begin{aligned} \overline{H_i(f_s)} &= \langle H_i(f_s) \rangle_s \\ \overline{M_i(f_s)} &= \langle M_i(f_s) \rangle_s \end{aligned} \right\} \quad (4-25)$$

Since, as compared to Equation 4-11:

$$M_i(f_s) + H_i(f_s) = - \sum_j p_{ij}(f_s) \log p_j \quad (4-26)$$

then;

$$\langle M_i(f_s) + H_i(f_s) \rangle_s = \overline{M_i(f_s)} + \overline{H_i(f_s)}$$

$$\begin{aligned}
&= - \sum_j p_{ij} \log p_j \\
&= M_1 + H_1
\end{aligned}
\tag{4-27}$$

Therefore, as before:

$$\overline{M_1(f_g)} + \overline{H_1(f_g)} = M_1 + H_1
\tag{4-28}$$

One of the major experimental problems is the proper selection of frequency intervals to evaluate. For some areas of the relative frequency spectrum a small change in interval size might lead to a large change in effectiveness; for other areas of the spectrum, however, changing the interval might have a negligible effect on effectiveness. These intervals will in general not be uniform over the spectrum and will be different for each word. Although this selection and evaluation appears difficult, it will lead to better category prediction.

4.2.1.3 Summary - Three ways of using word frequency information in documents to predict document categories have been indicated. Based upon earlier information theoretical concepts of document classification, this information can be evaluated in terms of its effectiveness as a clue to document categories. It is likely that the most effective clues would be found in relative frequency information--the ratio of clue word occurrence to the total number of document words. Once effective clues were found, they would be used exactly like the clues discussed in previous reports.

The measures of effectiveness, H_1 and M_1 , have been generalized; for each case the sum of the averages of the generalized H_1 and M_1 was always equal to $H_1 + M_1$. Thus, for the relative frequency case:

$$H_1 + M_1 = \overline{H_1(f_s)} + \overline{M_1(f_s)}$$

which seems to indicate that H_1 and M_1 produces a good average picture of word effectiveness.

The major difficulty with using word frequency information is the increase in computation required. In addition, where frequency intervals are used, the choice of intervals must be carefully determined. However, it is expected that category prediction would be much more accurate.

4.2.2 Non-Boolean Retrieval Processes

4.2.2.1 Introduction - In many cases Boolean search techniques are inadequate for retrieving information effectively. The objectives of this section, therefore, are:

- (a) To explicate the concept of non-Boolean retrieval.
- (b) To show the usefulness of non-Boolean retrieval processes.
- (c) To suggest the particular ways in which non-Boolean retrieval may be effected.

These concepts are presented in this section, even though they imply query capabilities, because of their dependence upon precise categorization.

Most of the presently operating retrieval systems assume that the ideal objective of the information search processes consists of retrieving

classes of documents corresponding to the descriptor function specified in the request. Thus, to every Boolean function of descriptors, there corresponds an identical function defined upon the set of classes of documents to which the descriptors are affixed. For example, of the retrieval request is $a \cdot b$ --where a, b are descriptors, and the dot, \cdot , signifies logical and--the class retrieval would be $D_a \cap D_b$; that is, the intersection of classes of documents designated by the descriptors 'a' and 'b' respectively. Yet, the effectiveness of retrieval procedures based upon this kind of correspondence depends upon an assumption that is not necessarily valid for all information retrieval systems. The assumption in question is: The documents fall into categories or classes unequivocally. In other words, the document belongs to a class of documents with either the probability 1 or probability 0. This section proves that the Boolean retrieval process will not be most efficient, in a certain sense, if the assumption is not true.

4.2.2.2 Inefficiencies in Boolean Retrieval - Before demonstrating the lack of effectiveness of Boolean retrieval, it would be desirable to consider situations in which probabilistic class assignment could be expected.

- (a) The Case of Many Users - A situation may occur where the views of users regarding membership of some documents in a certain category are divergent. Assume, for example, that there are 100 users, 5 categories, and 10 documents. Each user is asked to assign each document to one or more categories. Table 1 illustrates a possible set of choices. The numbers at the

TABLE 1. PROBABILISTIC ASSIGNMENT

CATEGORIES	DOCUMENTS									
	1	2	3	4	5	6	7	8	9	10
A	65	50	75	80	25	0	0	15	30	45
B	100	50	35	40	60	25	50	75	25	0
C	90	80	60	0	20	50	40	0	0	10
D	35	50	25	30	15	15	0	25	80	100

intersection of rows and columns indicate the probability of a document belonging to a certain category. Thus document No. 10 will belong to category D with probability 1, since all the users agree to place it there. On the other hand, the same document will have a probability of zero of belonging to category B; again, all the users agree to exclude it from this category. Since 45 percent of the users agreed to place document No. 10 in category A, it has been assigned a probability of .45.

- (b) Automatic Category Formation - Documents may be assigned to categories in accordance with an automatic procedure. This procedure may be intrinsically probabilistic in nature; that is, a document is assigned to a category with probability p depending upon the circumstances pertaining to the procedure of assignation.

Assume now that there is a collection of documents and a set of non-exclusive categories. Let p_{ij} be the probability that a document d_i

belongs to the category c_j . For the purpose of retrieval the boundaries of categories cannot remain indefinite. This restriction implies that a cutoff point for the probability should be established. A document d_i then is considered, for a particular retrieval query, to be within a category c_j if the probability of its being in the category p_{ij} is larger than the cutoff point value, σ . If all documents belonging to the intersection of two categories c_j and c_k are to be retrieved, then, assuming that the probabilities of documents belonging to categories are independent, the cutoff point $c_j \cdot c_k$ will be σ^2 . Thus it may be expected that some superfluous or extraneous documents will be retrieved.

From the point of view of retrieving the union of classes, there is a symmetrically opposite situation; some documents that are relevant will not be retrieved. If the cutoff point for the classes of documents defined by the descriptors a and b is again the probability value equal to σ , the probability of a document belonging to the class defined by the union $a \cup b$ will be $2\sigma - \sigma^2$. This quantity, however, is always greater than σ , since $\sigma \leq 1$; the proof is:

$$\begin{aligned} (2\sigma - \sigma^2) - \sigma &= \sigma - \sigma^2 \\ &= \sigma(1 - \sigma) \end{aligned} \tag{4-29}$$

which must always be positive.

This analysis proves that the standard of admissibility of a document to a class of retrieved documents cannot be maintained if the Boolean retrieval functions are used. The cutoff probability will be lowered in case of a retrieval criterion of logical intersection and

will be raised in the case of a union.

The question remains as to how the retrieval process is to be organized in order to preserve the same cutoff point for the results of retrievals upon any request. In continuing the analysis, it is necessary to formulate an explicit goal. Boolean retrieval has been proved inadequate in the sense of not preserving the criterion of admissibility. The problem, therefore, is to find a procedure that will permit the retrieval of classes of documents satisfying this criterion. The simplest system would calculate the probability of a document belonging to the category specified in the request; then the document would be accepted or rejected depending upon the value of calculated probability. However, a system of this nature may be uneconomical for the following reasons:

- (a) The system would be forced to scan documents with a low probability of belonging to a given descriptor. Such a procedure is uneconomical because the system must scan through a substantial portion of the document collection for every request.
- (b) The necessity of performing a computation for each document scanned to determine its probability of belonging to the class represented by request may increase the retrieval time beyond tolerable limits.

For reasons of economy, therefore, it may be useful to introduce an a priori fixed categorisation that would relieve the system of the necessity of scanning the documents with low probability values and performing the attendant computations.

This analysis has already shown that the formation of categories with a fixed probability cutoff point for a given descriptor implies that this criterion will not be preserved under general retrieval procedures,

which will generally specify more complex logical functions. If some concessions to economy are granted, the result will be a retrieval process that will omit some desirable documents and yield some undesirable ones. Within the framework of such a situation there may still be an optimum solution.

The basic premise is that the boundaries of descriptor extensions will be fixed a priori. At the same time these boundaries, the cutoff points, will be fixed in such a way as to maximize the value of the average retrieval process to the user. This premise does not necessarily include the restriction that a single cutoff point should be established for any descriptor extension; instead, the number of cutoff points should be established a priori, whatever that number might be.

The problem then resolves itself to:

- (a) Finding rational criteria for establishing what the user's value of retrieval procedures is.
- (b) Constructing a method for deriving the values of cutoff points that will optimize these criteria.

The rest of this section presents an analysis of these problems.

4.2.2.3 The Problem of Establishing Criteria for Determining User's Value of An Average Retrieval Procedure - With respect to any retrieval request the entire collection of documents may be divided into four subgroups:

- (a) The retrieved documents that are relevant.
- (b) The retrieved documents that are not relevant.
- (c) The unretrieved documents that are relevant.

(d) The unretrieved documents that are not relevant.

Since it was assumed that the descriptors are assigned to documents on a probabilistic basis, all four subgroups will be generally represented in any retrieval process.

Regardless of any special assumptions, it is clearly permissible to assert that as the number of documents in categories decreases, (a) and (d) increases and as the number of documents in categories (b) and (c) decreases, the value of the retrieved collection to the user will increase. Thus,

$$V = f_1\{I\} - f_2\{II\} - f_3\{III\} + f_4\{IV\} + K \quad (4-30)$$

where V is defined as the user value of the retrieved collection; f_1 , f_2 , f_3 , and f_4 are unspecified, monotonically increasing functions; and $\{I\}$, $\{II\}$, $\{III\}$, and $\{IV\}$ are the number of documents in the subclasses (a), (b), (c), and (d), respectively. K is defined as a constant that determines the minimal value for the user below which the retrieval is not justified under any circumstances.

For simplicity, replace f_1 , f_2 , f_3 , and f_4 by the constants α , β , γ , and δ , and set $K = 0$. The results of the discussion are not essentially modified by this simplification. Equation 4-30 then becomes:

$$V = \alpha\{I\} - \beta\{II\} - \gamma\{III\} + \delta\{IV\} \quad (4-31)$$

Since $K = 0$, the retrieval process should proceed as long as the increment of V , dV , is positive. That is, the process may select a group of documents with common probability characteristics (in relation to the request profile) and then investigate the change of V by including some additional

documents with lower probability characteristics. The question as to which documents will be retrieved is the problem of fixing the most advantageous values for the set $\{\sigma_1\}$ of cutoff points for the descriptor classes.

The appropriateness of replacing the functions f_1 , f_2 , f_3 , and f_4 by the constants α , β , γ , and δ rests upon the understanding of what factors could be responsible for non-linearity of the function V . Essentially there are two reasons why the function V should be non-linear. The first pertains to the economics of using documents; the other, to the problem of redundancy. In general, the efficiency with which the retrieved collection is used depends upon its size, even if the value of the individual documents in the collection is not prejudged. Nevertheless, since retrieval systems can be used in various ways, it is safe to assume that for many uses the relative emphasis placed upon the classes of retrieved and unretrieved documents remains unchanged. To the extent that this assumption is true, the fact that the function V depends upon class {IV}, the class of correctly unretrieved documents, helps to remedy the situation.

The second objection is more serious. Among the retrieved documents there may be a high degree of redundancy; in extreme cases the same amount of information may be covered more efficiently by a smaller number of documents. It is difficult, however, to decide whether or not redundancy is a linear function of the size of the retrieved collection. To answer this question adequately, it would be necessary to formalize the concept of redundancy among documents and then perhaps to formulate

theoretical prescriptions for procedures that would permit the system to retrieve the most efficient covering of the topic specified in the request. (This problem is a difficult task in itself and merits separate investigation.) Pending at least a crude formulation of the theory of redundancy, this discussion will be confined to the simplest assumption of linearity. Therefore, given the function V in the form of Equation 4-31, the first task is to find the set of σ -values, the cutoff points, that would maximize the user's value for an average retrieval process.

4.2.2.4 Method for Determining Cutoff Point Values - The following symbols will be used in this exposition:

N_T = total number of documents in the collection.

N_{i1} = total number of documents belonging to the descriptor i .

$n_i(p_K)$ = the number of documents containing the descriptor i within the probability interval centering around K .

$N_i(p)$ = the distribution function for the descriptor i defined on the probability values as a random variable.

$$N_i(p) = \int_0^p n_i(p) dp$$

\bar{p}_i = the average value of the probability for the descriptor i .

$$\bar{p}_i = \frac{1}{N_i} \int_0^1 n_i(p) p dp$$

$\bar{p}_i(\sigma)$ = the average probability value for the descriptor i in the probability interval between 0 and σ .*

*The normalization factor is N_i , not $N_i(\sigma)$.

$$\bar{p}_1(\sigma) = \frac{1}{N_1} \int_0^{\sigma_1} n_1(p) p \, dp$$

f_1 = the frequency with which the descriptor 1 is used.

s = the total number of descriptors.

σ_1 = the value of terminal probability defining the boundary of the class of documents belonging to descriptor 1.

Then, by definition:

$$\left. \begin{aligned} N_1(p) &= \int_0^p n_1(p) dp \\ \bar{p}_1 &= \frac{1}{N_1} \int_0^1 n_1(p) p \, dp \\ \bar{p}_1(\sigma) &= \frac{1}{N_1} \int_0^{\sigma_1} n_1(p) p \, dp \end{aligned} \right\} \quad (4-32)$$

In this discussion the descriptors are assumed to be independent. To facilitate computation, the number of documents in each class are assumed to be large enough and the subdivision into the probability brackets fine enough to permit integration techniques to replace summation.

The procedure for calculating the set of σ_1 's that will maximize V on pairs of descriptors is:

- (a) Calculate the numbers of documents for the four subclasses of documents that enter V for an unspecified σ_1 .
- (b) Obtain a general expression for V .
- (c) Obtain an expression for the expectation value for all V 's.
- (d) Differentiate the expression obtained under (c), and set the coefficient of differentials equal to zero in order to obtain a set of conditions for the maximum.
- (e) Solve the equations to obtain the values of the σ_1 's.

These steps can now be developed and expressed mathematically. The expression for the number of documents containing the descriptor i within the probability interval centering around p_{K_1} and the descriptor j within a probability interval centering around p_{K_2} is:

$$\frac{1}{N} n_i(p_{K_1}) \cdot n_j(p_{K_2}) dp_{K_1} dp_{K_2} \quad (4-33)$$

The probability of the document being within the two-dimensional probability interval centering around the values p_{K_1} and p_{K_2} is the product of the probabilities:

$$p(K_1, K_2) = p_{K_1} \cdot p_{K_2} \quad (4-34)$$

By using these equations, expressions can be calculated for the four classes of documents involved in the function V :

(a) Class I - The class of all correctly retrieved documents:

$$\{I\} = \int_{\sigma_1}^1 \int_{\sigma_j}^1 \frac{n_i(p_1) \cdot n_j(p_j)}{N} p_1 p_j dp_1 dp_j \quad (4-35)$$

(b) Class II - The class of all the incorrectly retrieved documents:

$$\{II\} = \int_{\sigma_1}^1 \int_{\sigma_j}^1 \frac{n_i(p_1) \cdot n_j(p_j)}{N} (1 - p_1 p_j) dp_1 dp_j \quad (4-36)$$

(c) Class III - The class of incorrectly unretrieved documents:

$$\{III\} = \int_0^{\sigma_1} \int_0^{\sigma_j} \frac{n_i(p_1) \cdot n_j(p_j)}{N} p_j p_j dp_1 dp_j \quad (4-37)$$

(d) Class IV - The class of all correctly unretrieved documents:

$$\{IV\} = \int_0^{\sigma_1} \int_0^{\sigma_j} \frac{n(p_1) \cdot n(p_j)}{N} (1 - p_1 p_j) dp_1 dp_j \quad (4-38)$$

The retrieval process proceeds until the predetermined cutoff point σ_1 for descriptor i and σ_j for descriptor j has been reached. To retrieve beyond this point will be detrimental, since on the average the increment in V caused by additional retrieval will be negative.

The four double integrals in Equations 4-35 through 4-38 can now be evaluated. For Equation 4-35:

$$\begin{aligned} \{I\} &= \int_{\sigma_1}^1 \int_{\sigma_j}^1 \frac{n(p_1)}{N} n(p_j) p_1 p_j dp_1 dp_j \\ &= \frac{1}{N} \int_{\sigma_1}^1 n(p_1) p_1 dp_1 \int_{\sigma_j}^1 n(p_j) p_j dp_j \\ &= \frac{1}{N} \{N_{1T} [\bar{p}_1 - \bar{p}_1(\sigma_1)] [N_{jT} (\bar{p}_j - \bar{p}_j(\sigma_j))]\} \end{aligned} \quad (4-39)$$

By using the definitions for $N_1(p)$ and N_{1T} , Equations 4-36 through 4-38 become:

$$\begin{aligned} \{II\} &= \int_{\sigma_1}^1 \int_{\sigma_j}^1 \frac{n_1(p_1) \cdot n_j(p)}{N} (1 - p_1 p_j) dp_1 dp_j \\ &= \frac{1}{N} \left\{ \begin{aligned} &[N_{1T} - N_1(\sigma_1)] [N_{jT} - N_j(\sigma_j)] \\ &- N_{jT} N_{1T} [\bar{p}_1 - \bar{p}_1(\sigma_1)] [\bar{p}_j - \bar{p}_j(\sigma_j)] \end{aligned} \right\} \end{aligned} \quad (4-40)$$

$$\begin{aligned}
\{\text{III}\} &= \int_0^{\sigma_1} \int_0^{\sigma_j} \frac{n_1(p_1) \cdot n_j(p_j) p_1 p_j dp_1 dp_j}{N} \\
&= \frac{1}{N} \cdot N_{1T} N_{jT} \bar{p}(\sigma_1) \bar{p}(\sigma_j) \quad (4-41)
\end{aligned}$$

$$\begin{aligned}
\{\text{IV}\} &= \int_0^{\sigma_1} \int_0^{\sigma_j} \frac{n_1(p_1) \cdot n_j(p_j)}{N} (1 - p_1 p_j) dp_1 dp_j \\
&= \frac{1}{N} [N_1(\sigma_1) N_j(\sigma_j) - N_{1T} N_{jT} \bar{p}(\sigma_1) \bar{p}(\sigma_j)] \quad (4-42)
\end{aligned}$$

By substituting Equations 4-39, 4-40, 4-41, and 4-42 into Equation 4-32, the function V becomes:

$$\left. \begin{aligned}
V_{ij} &= \frac{\alpha}{N} [N_{1T} N_{jT} [\bar{p}_1 - \bar{p}_1(\sigma_1)] [\bar{p}_j - \bar{p}_j(\sigma_j)]] \\
&\quad - \frac{\beta}{N} \{ [N_{1T} - N_1(\sigma_1)] [N_{jT} - N_j(\sigma_j)] \\
&\quad \quad - N_{jT} N_{1T} [\bar{p}_1 - \bar{p}_1(\sigma_1)] [\bar{p}_j - \bar{p}_j(\sigma_j)] \} \\
&\quad - \frac{\gamma}{N} N_{1T} N_{jT} \bar{p}(\sigma_1) \bar{p}(\sigma_j) \\
&\quad + \frac{\delta}{N} [N_1(\sigma_1) N_j(\sigma_j) - N_{1T} N_{jT} p_1(\sigma_1) p_j(\sigma_j)]
\end{aligned} \right\} \quad (4-43)$$

Now it is possible to find the values of σ_1 and σ_j that will maximize a specific V_{ij} . In general, however, the values σ_1' and σ_1'' obtained by solving for maxima in expressions V_{ij} and, say, V_{ik} will be different. This observation implies that we are looking for a set of values $\{\sigma_i\}$ that will maximize an average V_{ij} .

The average value of V_{ij} is, of course, its expected value:

$$E(V) = \frac{1}{2} \sum_{ij}^{ss} V_{ij} f_i f_j \quad (i \neq j) \quad (4-44)$$

and this function will have to be maximized. The differential of Equation 4-44 is:

$$\left. \begin{aligned} dE &= \frac{1}{2} \sum_{i,j}^{ss} f_i f_j \left[\frac{\partial V_{ij}}{\partial \sigma_i} d\sigma_i + \frac{\partial V_{ij}}{\partial \sigma_j} d\sigma_j \right] \quad (i \neq j) \\ \text{or} \quad dE &= \sum_i^s f_i \left[\sum_j^s f_j \frac{\partial V_{ij}}{\partial \sigma_i} \right] d\sigma_i \end{aligned} \right\} \quad (4-45)$$

which implies the following condition for a maximum:

$$\sum_j^s f_j \frac{\partial V_{ij}}{\partial \sigma_i} = 0 \quad (i = 1, 2, \dots, s) \quad (4-46)$$

The partial derivatives $\partial V_{ij}/\partial \sigma_i$ in Equation 4-46 can be computed by using Equations 4-39, 4-40, 4-41, 4-42, and 4-43:

$$\frac{\partial \{I\}}{\partial \sigma_i} = -\frac{N_{iT}}{N} \{[\bar{p}_j - \bar{p}_j(\sigma_j)] [-\sigma_i n_i(\sigma_i)]\} \quad (4-47)$$

$$\begin{aligned} \frac{\partial \{II\}}{\partial \sigma_i} &= \frac{1}{N} [-N_{jT} + N_j(\sigma_j)] [n_i(\sigma_i)] \\ &\quad + \frac{N_{jT}}{N} [\bar{p}_j - \bar{p}_j(\sigma_j)] [\sigma_i n_i(\sigma_i)] \end{aligned} \quad (4-48)$$

$$\frac{\partial \{III\}}{\partial \sigma_i} = -\frac{N_{jT}}{N} \bar{p}(\sigma_j) \sigma_i n_i(\sigma_i) \quad (4-49)$$

$$\frac{\partial \{IV\}}{\partial \sigma_i} = \frac{1}{N} [N_j(\sigma_j) n_i(\sigma_i) - N_{jT} \bar{p}(\sigma_j) \sigma_i n_i(\sigma_i)] \quad (4-50)$$

Performing the summations in Equation 4-46 on Equations 4-47 through 4-50 results in:

$$\sum_j f_j \frac{\partial \{I\}}{\partial \sigma_i} = -\frac{1}{N} \sigma_i n_i(\sigma_i) \sum_{j=1}^s f_j N_{jT} [\bar{p}_j - \bar{p}_j(\sigma_j)] \quad (4-51)$$

$$\begin{aligned} \sum_j f_j \frac{\partial \{II\}}{\partial \sigma_1} &= \frac{1}{N} [n_1(\sigma_1)] \sum_{j=1}^s -N_{jT} + N_j(\sigma_j) f_j \\ &\quad + \frac{\sigma_1 n_1(\sigma_1)}{N} \sum_{j=1}^s [\bar{p}_j - \bar{p}_j(\sigma_j)] f_j N_{jT} \end{aligned} \quad (4-52)$$

$$\sum_j \frac{\partial \{III\}}{\partial \sigma_1} = \frac{1}{N} \sigma_1 n_1(\sigma_1) \sum_{j=1}^s f_j N_{jT} \bar{p}(\sigma_j) \quad (4-53)$$

$$\begin{aligned} \sum_j \frac{\partial \{IV\}}{\partial \sigma_1} &= \frac{n_1(\sigma_1)}{N} \sum_{j=1}^s f_j N_j(\sigma_j) \\ &\quad - \frac{1}{N} \sigma_1 n_1(\sigma_1) \sum_{j=1}^s f_j N_{jT} \bar{p}(\sigma_j) \end{aligned} \quad (4-54)$$

Therefore, the following equations can be solved for the σ_1 's:

$$\left. \begin{aligned} & - \frac{\alpha}{N} \sigma_1 n_1(\sigma_1) \sum_{j=1}^s f_j N_{jT} [\bar{p}_j - \bar{p}_j(\sigma_j)] \\ & - \frac{\beta}{N} n_1(\sigma_1) \sum_{j=1}^s [-N_{jT} + N_j(\sigma_j)] f_j \\ & - \frac{\beta}{N} \sigma_1 n_1(\sigma_1) \sum_{j=1}^s [\bar{p}_j - \bar{p}_j(\sigma_j)] f_j N_{jT} \\ & - \frac{\gamma}{N} \sigma_1 n_1(\sigma_1) \sum_{j=1}^s f_j N_{jT} \bar{p}(\sigma_j) \\ & + \frac{\delta}{N} n_1(\sigma_1) \sum_{j=1}^s f_j N_j(\sigma_j) \\ & - \frac{\delta}{N} \sigma_1 n_1(\sigma_1) \sum_{j=1}^s f_j N_j \bar{p}(\sigma_j) = 0 \end{aligned} \right\} \quad (4-55)$$

for $i = 1, 2, \dots, s$; where $i \neq j$.

In order to get some insight into the nature of the solution, set $\gamma = \delta = 0$; i.e., the function V depends only upon classes {I} and {II}. In this case, Equation 4-55 is simplified to:

$$\left. \begin{aligned} & -\frac{\alpha}{N} \sigma_i n_i(\sigma_i) \sum_{j=1}^s f_j N_{jT} [\bar{p}_j - \bar{p}_j(\sigma_j)] \\ & -\frac{\beta}{N} n_i(\sigma_i) \sum_{j=1}^s [-N_{jT} + N_j(\sigma)] f_j \\ & -\frac{\beta}{N} \sigma_i n_i(\sigma_i) \sum_{j=1}^s [\bar{p}_j - \bar{p}_j(\sigma_j)] f_j N_{jT} = 0 \end{aligned} \right\} \quad (4-56)$$

for $i = 1, 2, \dots, s$; where $i \neq j$. After rearranging and dividing by the common factor, $n_i(\sigma_i)/N$, Equation 4-56 becomes:

$$\sigma_i = \frac{-\beta \sum_{j=1}^s f_j [N_j(\sigma_j) - N_{jT}]}{(\alpha + \beta) \left[\sum_{j=1}^s f_j N_{jT} [\bar{p}_j - \bar{p}_j(\sigma_j)] \right]} \quad (4-57)$$

for $i = 1, 2, \dots, s$; where $j \neq i$.

The solution of Equation 4-57 for different i 's, say i_1 and i_2 , are almost identical. The solutions differ only by the absence in the summation on the right hand side of the term corresponding to i . To demonstrate this point more clearly, redefine the summations in Equation 4-57 as:

$$\left. \begin{aligned} & \sum_{j=1}^s f_j [N_j(\sigma_j) - N_{jT}] = g(N) \\ & \sum_{j=1}^s f_j N_{jT} [\bar{p}_j - \bar{p}_j(\sigma)] = g(\bar{p}) \end{aligned} \right\} \quad (4-58)$$

Then, by inserting Equation 4-58 in Equation 4-57 and adding back the term for $j = i$:

$$\sigma_i = \frac{-\beta g(N) + \beta f_i[N_i(\sigma_i) - N_{iT}]}{(\alpha + \beta)g(\bar{p}) - (\alpha + \beta)N_{iT} f_i[\bar{p}_i - \bar{p}_i(\sigma_i)]} \quad (4-59)$$

Since the two g terms represent summations over all values of j , they are identical for all σ_i 's. Now, if s is large, the terms $f_i[N_i(\sigma_i) - N_{iT}]$ and $f_i[\bar{p}_i - \bar{p}_i(\sigma_i)]$ are small compared with $g(N)$ and $g(\bar{p})$, respectively. The reason is that with the large number of descriptors, thus a large s , the weights f_i , which represent the frequency of usage of descriptors, are all small fractions of the order $1/s$. Therefore, the values of σ_i 's are approximately equal. If σ_i is multiplied by the denominator of the expression on the right hand side of Equation 4-59 and summed over all i , then:

$$\left. \begin{aligned} s\sigma_i(\alpha + \beta)g(\bar{p}) - (\alpha + \beta)\sigma_i \sum_{i=1}^s N_{iT} f_i[\bar{p}_i - \bar{p}_i(\sigma_i)] \\ = -s\beta g(N) + \beta \sum_{i=1}^s f_i[N_i(\sigma_i) - N_{iT}] \end{aligned} \right\} \quad (4-60)$$

From the definitions of the two g functions, Equation 4-60 becomes:

$$s\sigma_i(\alpha + \beta)g(\bar{p}) - (\alpha + \beta)\sigma_i g(\bar{p}) = -s\beta g(N) + \beta g(N)$$

or

$$(s - 1)(\alpha + \beta)g(\bar{p})\sigma_i = (-s + 1)\beta g(N)$$

or

$$\boxed{\sigma_i = -\frac{\beta g(N)}{(\alpha + \beta)g(\bar{p})}} \quad (4-61)$$

which is equivalent to Equation 4-57.

The minus sign in Equation 4-61 occurs because $g(N)$ is inherently negative. Each term in the summation for $g(N)$ is negative. Since $N(\sigma)$ is a monotonically increasing function of σ , it is now possible to interpret the meaning for the value of σ , established in Equation 4-61.

It is apparent that $-g(N)$ represents the average or expected number of retrieved documents. On the other hand, each term of $g(\bar{p})$ represents a product of the average probability of retrieved documents times the size of the descriptor group normalized by the frequency of usage of this descriptor. Thus the $g(\bar{p})$ function expresses the average number of retrieved documents properly belonging to the average descriptor weighed by its frequency of occurrence. It is thus seen that the optimum σ , expressed by Equation 4-61, is a function of the constants α and β , which express the relative importance attached to the correctly and incorrectly retrieved documents; the optimum σ is also a function of two averages--namely, $g(N)$ and $g(\bar{p})$.

It is evident that the higher the value of β --i.e., the importance attached to incorrectly valued documents--the higher will be the value of σ . And as σ increases, fewer documents will be retrieved. On the other hand, the higher the value of α --i.e., the importance attached to the correctly retrieved documents--the lower will be the value of σ . For lower values of σ more documents will be retrieved. The function $-g(N)$ decreases with the increment of value of σ , and so does $g(\bar{p})$. When $\sigma = 0$:

$$\left. \begin{aligned} g(N) &= - \sum_{j=1}^s f_j N_{jT} \\ g(\bar{p}) &= \sum_j^s f_j N_{jT} \bar{p}_j \end{aligned} \right\} \quad (4-62)$$

and when $\sigma = 1$.

$$g(N) = g(\bar{p}) = 0 \quad (4-63)$$

Thus, at $\sigma = 0$:

$$\frac{-\beta g(N)}{(\alpha + \beta)g(\bar{p})} = \frac{\sum_{j=1}^s f_j N_{jT}}{\sum_j^s f_j N_{jT} \bar{p}_j} \quad (4-64)$$

To evaluate the expression for $\sigma = 1$, L'Hopital's rule must be used because of the indeterminacy of 0/0:

$$\left. \begin{aligned} \frac{g(N)}{g(\bar{p})} &\rightarrow \frac{-g'(N)}{g'(\bar{p})} \text{ as } \lim g(N) = \lim g(\bar{p}) = 0 \\ -g'(N) &= - \sum_{j=1}^s f_j n_j(\sigma) \\ g'(\bar{p}) &= - \sum_{j=1}^s f_j N_{jT} \sigma n_j(\sigma) \end{aligned} \right\} \quad (4-65)$$

Thus, at $\sigma = 1$:

$$\frac{\beta g(N)}{(\alpha + \beta)g(\bar{p})} = \frac{\beta}{\alpha + \beta} \frac{\sum f_j n_j(\sigma)}{\sum f_j N_{jT} n_j(\sigma)} \quad (4-66)$$

Now, if the largest N_{jT} within $N_{j\max}$ is factored out of the denominator:

$$\frac{\beta}{\alpha + \beta} \frac{\sum f_j n_j(\sigma)}{N_{j\max} \sum f_j \frac{N_{jT}}{N_{j\max}} n_j(\sigma)} \sigma = 1 \quad (4-67)$$

Since $N_{jT}/N_{j\max} < 1$ for all j , it is clear that:

$$\frac{\sum f_j n_j(\sigma)}{\sum f_j \frac{N_{jT}}{N_{j\max}} n_j(\sigma)} < 1$$

Thus, for $\sigma = 1$, Equation 4-61 can only be satisfied if:

$$\frac{\beta}{(\alpha + \beta)N_{j\max}} > 1$$

But this result is an impossibility. This fact demonstrates that it is never advantageous to admit only the documents that belong to a class specified by a descriptor with certainty.

The formulas derived in this analysis pertain to joint retrieval on two descriptors. Similar derivations, although somewhat more complex, can be carried out for the arbitrary joint retrievals on k descriptors. The task of deriving these formulas will be continued in subsequent research activity.

Beyond joint retrievals there looms a question of retrievals specified in a request by an arbitrary Boolean function. Such problems may be handled by breaking up the arbitrary Boolean function into a canonical form of disjunction of conjunctions. All that is now necessary are formulas for calculating cutoff probabilities for disjunctions. This problem will also be handled as a part of future activity.

4.2.2.5 Conclusions - It is now possible to outline the general features of a non-Boolean retrieval system. To each descriptor there will correspond a collection of classes of documents instead of a unique class of documents. Each class will be determined by a different cutoff point σ . For each document, there will be two types of cutoff points, disjunctive and conjunctive. Within each of these categories an individual σ will have its value determined in accordance with the type of joint retrieval it is scheduled to participate in. Thus there will be one cutoff point for the conjunction of two descriptors, another one for conjunction of three, etc. The same principle holds for the cutoff points for disjunctive retrievals. Any incoming request will be transformed into convenient canonical form; for example, a disjunction of conjunctions. The appropriate cutoff points will then be selected and retrieval effected.

In order to calculate the cutoff points, certain parameters are required. These parameters can be obtained by requiring the system to perform bookkeeping operations which will supply the required data. Essentially, the kind of statistical data necessary for the calculation of the cutoff points is:

- (a) $n_i(p)$ = the "density" of documents pertaining to a given descriptor for a given probability interval.
- (b) $\bar{p}_i(\sigma)$ = the average probability value of a document belonging to the descriptor i as a function of a cutoff point.
- (c) $N_i(\sigma)$ = the total number of documents belonging to the descriptor i as a function of σ .

The most fundamental of the three types of data is (a), since (b) and (c) can be calculated from it.

4.2.2.6 Priorities for the Future - At this time the most

important extensions of this task appear to be:

- (a) The derivation of values of σ for the joint retrieval of products of arbitrary number of descriptors.
- (b) The derivation of values of σ for the joint retrieval of logical sums of the arbitrary number of descriptors.

The activity at the next in order of precedence will involve:

- (a) The evaluation of errors arising out of approximations used in the derivations.
- (b) The consideration of modifications arising out of the removal of the assumption of the independence of descriptors.
- (c) Considerations of an economic nature pertaining to the costs involved in the implementation of the non-Boolean retrieval systems for different types of applications.

4.3 QUERY CAPABILITIES

Work performed in this area deals with an approach to the generation of extracts or abstracts and with the problem of redundancy. The relaxation of limitations on description is dealt with indirectly in the preceding material on non-Boolean retrieval.

4.3.1 An Approach to a Criterion for Automatically Generated Extracts -

Automatic extracting was originally described by Luhn [1] some time ago. While he refers to the end products of his process as abstracts, they are more accurately characterized as extracts of what are hopefully the more central, critical, or descriptive sentences in a document. Luhn's technique is purely statistical. Sentences are selected for extracting on the basis of two related facts about their word content:

- (a) The relative frequency of the words in the sentence, except for common words.

- (b) The distance between high frequency words in the sentence, based upon the number of intervening non-clue words.

While Luhn presents a rather vague theoretical rationale for the validity of such an approach, there is no attempt to justify it in detail, except on the grounds that it can produce useful extracts. No attempt is made to show whether extracts generated by any other technique are more or less useful. Recently Guillian, et al [2], at Arthur D. Little have proposed a technique for incorporating syntactic information into the distance measure in order to make the technique more useful.

There seem to be two things lacking in this approach to automatic abstracting or extracting:

- (a) A lack of any criterion or perhaps of multiple criteria, depending on the context in which the extract is to be used, for determining the adequacy of any given extract or extracting scheme.
- (b) A lack of understanding of the fundamental processes involved in human abstracting, extracting, condensation, or perception of statement saliency in a longer argument or presentation.

It would seem that a combination of the approach of Newell and Simon [3] to the simulation of cognitive processes--theorem proving and problem solving more generally--and the approach of Maron [4] to the automatic classification of documents might be appropriate. While each of these studies is well known, it might be appropriate to indicate briefly which aspects of their methodology are relevant to alleviating the two shortcomings in present automatic extracting systems.

Newell, et al, in order to simulate cognitive functioning, first used a method of observation and introspection to gain insight into the

method by which humans proved logic theorems. In the context of information retrieval the major emphasis is on useful extraction rather than on the simulation of human extraction. It may nevertheless pay to observe human extracting behavior in order to develop more useful algorithms for obtaining automatic extracts.

The work of Maron and Kuhns has already been described in previous reports. It involved the use of human classification of a set of items as a criteria for automatic classification. The automatic classification, however, was not based on the unknown techniques of the human classifiers. The automatic algorithm was based rather upon purely statistical features of some of the classified documents. Human classification was also available, however, to provide the criteria for checking the adequacy of the automatic algorithm once it was derived.

In the case of automatic extracting both of these techniques might prove useful. That is, the use of observation and introspection would help alleviate the difficulty caused by the lack of understanding of human functions and allow for the development of more rational extracting algorithms. Perhaps these techniques could be ultimately extended to abstracting per se. The records of humanly generated extracts could be used as a criterion for evaluating the adequacy of various automatic algorithms. The latter would alleviate the difficulty caused by the non-existence of suitable criteria.

The paradigm for such research and development would be as follows:

- (a) A series of documents, either larger texts or shorter articles for research convenience, would be selected for extracting.
- (b) Ground rules for desired extracts would be developed; e.g.:
 - (1) How long should each extract be? Should it be some fixed proportion of the total document?
 - (2) What sentential units should be extracted? Whole sentences only? Parts of sentences? Parts that can be recombined to form larger sentences?
 - (3) What is the focal purpose of the extract? To extract as much factual information as possible within the limits imposed by the length of an extract? To characterize the document as well as possible in order that the reader might know what information it contains? Both of these?
 - (4) What information or techniques may be used in generating the extract? Anything that occurs to the user based upon his total knowledge? Anything based on the explicit and implicit content of the document? Only explicit content? Only rigorously formulated rules?
- (c) The documents would then be subjected to human extracting using instructions based upon the ground rules.
- (d) A portion of the humanly extracted documents would be carefully subjected to introspective report and an analysis of the implicit rules followed in extracting.
- (e) Based on this analysis, one or several automatic algorithms would be developed for achieving essentially the same extracts from readily treated information in the documents. For the sake of generality, an attempt would also be made to incorporate those rules manifest in introspective protocols that could be handled by computers.
- (f) Measures of correspondence between humanly and automatically generated extracts would then be developed.
- (g) Finally, the automated techniques would be applied to the remaining documents in the sample and the extracts generated would be validated against the criterion of the human extracts already available.

While this approach depends upon research and development strategies already developed by others, its application to the information retrieval

problem is unique. It would probably be unwise to embark on a specific program of this kind in the remaining part of this project, but further research along these lines seems unwarranted.

4.3.2 The Problem of Redundancy in Information Retrieval Systems

4.3.2.1 Introduction - Redundancy in the information retrieval processes occurs whenever the retrieved data is duplicated. To avoid redundancy is important, not only for the rather obvious economic reason, but also for operational and logical reasons. Theoretical considerations pertaining to the nature of measures for removing redundancy will be best understood within the context of a more detailed discussion of the undesirability of duplication from these three points of view.

4.3.2.2 Economic Point of View - For some types of information retrieval systems the cost of retrieval may become prohibitively high, especially if all the data pertaining to the request profile is retrieved.

The use value of the information contained in the retrieved data may be drastically reduced by the existence of redundant material. Effectively the user of the data is swamped by repetitious information.

4.3.2.3 Operational Point of View - Many information retrieval systems enter into larger systems as component units. The retrieved data may form an input to other processes such as control, command and control, or real-time monitoring. The occurrence of redundant material may not only reduce the efficiency of the functioning of the system, but may also affect the outcome of the processes to which the retrieved data forms an

input. For example, imagine a system that is required to perform some statistical tabulations on the incidence of car accidents among various population groups. Furthermore, assume that the reports on automobile accidents are incoming from diverse sources so that some accidents may be reported more than once. Under such conditions it will be necessary, in order to obtain valid results, to introduce some filtering stage that will prevent or eliminate duplication. Estimates of the reliability of the results obtained will in general depend upon the effectiveness of the filtering stage. The removal of data redundancy is thus vital to the satisfactory performance of the system as a whole.

4.3.2.4 Logical Point of View - In the process of decision making the origin of the data may be as relevant to the decision as its content. It is even conceivable that the existence of large redundancy in the collected data may be one of the important factors influencing the nature of the decision. In other words, the decision process may be dependent on the manner in which the data is presented. As an example, imagine a system whose task it is to solve transportation-routing problems. The kind of solution employed may well depend upon the complexity of a particular problem. If the particular transportation network contains many nodes, the system will use one type of an algorithm; if it contains few nodes, then another.

Determining the nature of the problem may depend upon sampling of data; thus inaccuracies will arise if the data contains a large amount of redundancy. Such a situation is particularly prone to arise if the

system schedules its own operations and batches many problems together.

Considering several ways in which the concept of redundancy is implicated in the information retrieval processes, one observes a basic dichotomy:

- (a) Some of the redundancy problems require the exact scrutiny of the individual data items. If data items are conventionally thought of as documents, then a sort of redundancy map could be obtained by indicating the relationship with respect to the redundancy of each document to every other document in the collection. The simplest kind of relation between documents with respect to redundancy is that of inclusion; that is, one document may express everything that another document expresses with respect to a given topic. Another possible relation, although a less simple one, is that of overlap. A document may partially express the content of another document with respect to a given topic with some numerical measure of the partial covering.
- (b) It may be possible or/and desirable to handle the problem of reducing redundancy on an aggregate level. The distinguishing feature of this approach is the statistical handling of information contained in the documents. It is important to remember that, since the primary concern is redundancy, the basic measure of information must be relative rather than absolute. That is, such a measure when applied to a document should be able to

determine the expected number of documents rendered superfluous by the document in question; alternatively, the measure should indicate how many documents render a given document superfluous.

Usually a document will cover a number of topics. In general, it must be expected that the redundancy measure will not be evenly distributed among all the topics that a given document deals with. Thus with respect to one topic a document may be highly unique, whereas with respect to another, highly redundant. Whether or not it is advisable to average the redundancy measure over all topics or handle them separately is a question that may be decided only after a more detailed and rigorous study. It is also possible that this question admits no unique answer, since information retrieval systems are highly differentiated with respect to their functional characteristics.

It would be incorrect to assume that this dichotomy represents two alternative approaches. It is quite unrealistic to expect that an exhaustive redundancy map comprising the detailed breakdown of all relations among all documents individually is feasible. Practically, some sort of statistical approach is necessary. It is necessary, however, to demand that any statistical averages employed to reduce redundancy capture the true statistical properties of a system based upon the requirements for a redundancy map.

4.3.2.5 Conclusion - In conclusion, two tentative examples of redundancy measures are given:

- (a) Each document is characterized by a set of numbers expressing the percentage of documents containing more, or less, information concerning a given topic.
- (b) Each document is characterized by a set of numbers expressing the additional contribution that the document would make to the given topic, assuming the average number of documents already retrieved.

4.4 PROCESSING CAPABILITIES

Work in this area has been primarily concerned with organization and search procedures. No new progress has been made on the problem of associative techniques.

4.4.1 Comparative Analysis of Some File Organisations

4.4.1.1 Introduction - This section contains a discussion of a number of file organisations that may be suitable for the retrieval of documents or other items of information. The exposition largely follows the order of mathematical development rather than some didactic organisation for easily communicating the results. This method of exposition is used because it is impossible in work of this kind to know at the beginning where fruitful mathematical analysis will lead.

For each file structure considered, expressions are derived for the average or expected values of the number of items and the subject or category headings examined to retrieve a single item, known to be in the file, in response to a request. The file organizations are then compared and evaluated in terms of these expected values for a wide range of file sizes. To aid in the comparison, variances are derived and plotted.

Three different types of file organizations or structures will be compared. They are:

- (a) Single-level subject headings.
- (b) Hierarchical trees of items.
- (c) Hierarchical trees of subject headings.

The first type consists of a single level of unrelated subject headings or category names under which items are grouped or filed in a linear sequence. An alphabetical card file is an example. The subject headings in this example are simply the letters of the alphabet.

The second type of file organization is a multi-level tree of items that are connected by the tree structure. This connectivity does not necessarily imply, however, a corresponding logical relation among these items.

The tree of subject headings, on the other hand, is a multi-level categorisation of subject headings where each heading is divided into two or more sub-headings down to the lowest level of detail. The tree of subject headings is intended to imply the logical relation among them. In this type of file it is assumed that the items are filed in a linear sequence or in a hierarchical tree under the last row of headings.

More than one way of searching the nodes of a tree will be used. Further subdivisions of the three types of file organizations will be discussed in the following detailed analysis. Trees of both items and subject headings will be considered, in various cases, in the section on hierarchical trees. First, however, single level subject headings will

be analyzed. This analysis will include the case of a sequentially ordered file which, when searched logarithmically, makes the transition between single level subject headings and hierarchical trees one of generalizing a special case.

For each type of file structure a mathematical expression can be derived for the expected number of headings and items searched and examined in order to locate a single item in the file. Some simplifying assumptions will be made to keep the mathematics relatively uncomplicated. Similar expressions can be derived, however, under less restrictive assumptions.

4.4.1.2 Single Level Subject Headings - Suppose there are s subject headings. It is assumed that the subject heading under which the item is to be found is supplied with the request. It is further assumed for the sake of simplicity that the items in the file are evenly distributed under the subject headings. That is, it is equally likely that any subject heading and any item under a subject heading will be requested and each subject heading will have the same number of items filed under it. The probability p_1 of searching one subject heading is:

$$p_1 = \frac{1}{s} \quad (4-68)$$

The probability of searching two subject headings to find the requested one is:

$$p_2 = \frac{s-1}{s} \cdot \frac{1}{s-1} = \frac{1}{s} \quad (4-69)$$

Similarly:

$$p_i = \frac{1}{s} \quad (4-70)$$

The expected number $E(i)$ of subject headings searched is:

$$\begin{aligned} E(i) &= \sum_{i=1}^s i \frac{1}{s} \\ &= \frac{1}{s} \frac{s(s+1)}{2} \end{aligned} \quad (4-71)$$

or

$$E = \frac{s+1}{2} \quad (4-72)$$

The number of items N_s under each subject heading is:

$$N_s = \frac{N}{s} \quad (4-73)$$

By an argument analogous to that for subject headings, the expected number $E(i)$ of items searched is:

$$\begin{aligned} E(i) &= \sum_{i=1}^{N_s} i \frac{1}{N_s} \\ &= \frac{N + s}{2s} \end{aligned} \quad (4-74)$$

The expected number of items and subject headings searched for in a linear file is then:

$$\begin{aligned} E &= \frac{s+1}{2} + \frac{N+s}{2s} \\ &= \frac{1}{2}(s + N/s + 2) \end{aligned} \quad (4-75)$$

A file of items arranged sequentially by some ordering rule--
e.g., a file of part or drawing numbers or any other numbered or ordered

items--can be arranged and searched by the method of subject headings previously described. Another method of search is the following: Go to the middle of the file. Compare the item requested with the item there. A decision can then be made on the basis of the ordering of the items as to whether the item sought is in the first (lower) half of the file or in the second (higher) half. Whichever half it is in, go to the middle of that half and repeat the procedure. This process is continued until the item is located. The process of going to the middle of any portion of the file will be called a cut. Since a single file item is examined for each cut, the expected number of cuts is equal to the expected number of file items which will be examined. This method is called the Binary Logarithmic search.

Consider a file of N items. By the search procedure just described, the number of items N_j that can possibly be retrieved on the first cut is 1, on the second cut, 2; and in general on the j^{th} cut:

$$N_j = 2^{j-1} \quad (4-76)$$

The maximum number of cuts n required to retrieve any item whatsoever in the file can be determined from Equation 4-76 as follows:

$$\begin{aligned} N &= \sum_{j=1}^n N_j \\ &= \sum_{j=1}^n 2^{j-1} \\ &= 2^n - 1 \end{aligned} \quad (4-77)$$

Solving Equation 4-77 for n gives:

$$n = \log_2(N + 1) \quad (4-78)$$

The origin of the name logarithmic search is obvious from Equation 4-78.

It is evident from Equation 4-76 that the probability p_j of retrieving the correct item in response to a given random request on the j^{th} cut is:

$$p_j = \frac{2^{j-1}}{N} \quad (4-79)$$

The expression for the expected number of cuts j (or, equivalently, the number of items examined) is:

$$E = \sum_{j=1}^n j \frac{2^{j-1}}{N} \quad (4-80)$$

where n is obtained from Equation 4-78. The series in Equation 4-80 is the derivative of a geometric progression, and the expression for its sum can be obtained by differentiating the expression for the sum of a geometric progression with a finite number of terms. This procedure yields the following expression for E :

$$E = \left[\frac{N+1}{N} \right] \log_2(N+1) - 1 \quad (4-81)$$

4.4.1.3 Hierarchical Trees - Only regular rooted trees will be considered for hierarchical trees. A tree is rooted if all its branches are connected ultimately to a single node (the root). A tree is regular if the number of branches k emanating from each node is a constant. Another way of thinking of this file structure is that every heading or grouping of the file organization is divided into the same number of subheadings.

Four cases of retrieving items from trees will be considered. These cases are designated I to IV, respectively.

4.4.1.3.1 Case I - In this case the tree is considered as a hierarchy composed entirely of file items, each of which is equally likely to be the answer to a given random request. Hence, retrieving a given node will be considered as providing a single-item response. The level of the node then represents the generality of the response, which is presumably related directly to the generality of the request. The node provided as a response can be considered as the name or term or descriptor for all the nodes at lower levels of the tree that are connected to the node provided as a response. If the node is a category name, all the connected nodes--the items in the category--could be provided as part of the response. It is assumed that the tree is indexed; that is, each node of the tree contains indexes of the nodes on the next lower level connected to it. It is also assumed that these indexes are sufficient to ascertain which node to examine at the next level. Thus only one node is examined at each level searched.

If each node of the tree contains indexes that are identifiers of the nodes at the next level at the end of the branches emanating from it, then by examining a given node a decision can be made as to which node to examine at the next level. Searching a tree of this type is a generalization of the binary logarithmic search. For example, consider a regular binary tree; that is, $k = 2$. Examining the first node, the root, is analogous to going to the middle of the file. There are two

nodes at the next level. Selecting one is analogous to going to the middle of the lower half of the file; selecting the other is equivalent to going to the middle of the upper half of the file. The generalization of this process for larger integral values of k is obvious. The mathematics is analogous to the binary logarithmic search.

The number of levels L to be examined in order to guarantee the retrieval of any item in a regular tree of order k is:

$$L = \log_k [(k - 1)N + 1] \quad (4-82)$$

The expected number of items examined becomes:

$$\begin{aligned} E &= \frac{1}{N} \sum_{j=1}^L j k^{j-1} \\ &= \left[\frac{(k - 1)N + 1}{(k - 1)N} \right] \log_k [(k - 1)N + 1] - \frac{1}{k - 1} \end{aligned} \quad (4-83)$$

where L is determined from Equation 4-82. Thus Equations 4-78 and 4-81 are merely special cases of Equations 4-82 and 4-83, respectively, for regular binary trees.

4.4.1.3.2 Case II - In this case only the nodes at the bottom level of the tree represent file items. It is assumed that each such node represents a group of file items. Thus a search consists of tracing a path through the tree to one node at the bottom and searching the items filed under that node to provide a single file item as a response. Again, it is assumed that each node is equally likely to be the answer. If this case is restricted to regular trees with no method of indexing or determining which connected node at the next level is the correct one, then this case generalizes the simple subject heading file to a multi-level

subject heading or classification file. Only non-indexed trees will be considered in this case. A non-indexed tree is one that has no mechanism for selecting the proper node at the next lower level without examining the nodes at that level connected to the node at which the searcher is presently located.

Assume there are s nodes or subject headings on a regular tree of order k . Then let there be N file items listed under the bottom nodes and assume that the file items are evenly distributed among these nodes. Assume also that there are L levels of nodes in the tree.

Since the only nodes searched at each level are those connected to the node selected at the next higher level, the probability p_j of finding the desired subject heading at a given node is:

$$p_j = \frac{1}{k} \quad (4-84)$$

Therefore, the expected number of nodes examined at any level j , except the first level or the root node* where the expected number is 1, is:

$$\begin{aligned} E_j(i) &= \sum_{i=1}^k \frac{1}{k} \\ &= \frac{k+1}{2} \end{aligned} \quad (4-85)$$

where $2 \leq j \leq L$. Hence, the expected number of nodes examined for the entire tree including the root node is:

*It is assumed that this node is examined to identify the tree and locate the nodes at the second level.

$$E_s = \left[\frac{k+1}{2} \right] (L-1) + 1 \quad (4-86)$$

The required number of levels L in the tree is determined by k and s , and is obtained from Equation 4-82, which gives:

$$L = \log_k [(k-1)s + 1] \quad (4-87)$$

Substituting Equation 4-87 into Equation 4-86 and simplifying:

$$E_s = \left[\frac{k+1}{2} \right] \log_k [(k-1)s + 1] + \frac{1-k}{2} \quad (4-88)$$

At this stage, no file items have been examined. Equation 4-88 gives the expected number of subject headings examined to find the heading at the lowest level under which the file item sought is listed. Therefore, the file items under that heading must now be examined. The number of items N_s filed under a given subject heading is:

$$N_s = \frac{N}{s_L} \quad (4-89)$$

where s_L is the number of subject headings, or nodes, at the lowest level of the tree. This sequence is a simple linear file like the first one examined. The expected number of file items searched E_n is then:

$$\begin{aligned} E_n(i) &= \sum_{i=1}^{N_s} \frac{1}{N_s} \\ &= \frac{N_s + 1}{2} \end{aligned} \quad (4-90)$$

The number of nodes s_j at level j of a regular tree of order k is given by:

$$s_j = k^{j-1} \quad (4-91)$$

therefore;

$$s_L = k^{L-1} \quad (4-92)$$

Substituting Equation 4-87 into Equation 4-92 yields:

$$s_L = \frac{(k-1)s + 1}{k} \quad (4-93)$$

and from Equations 4-89 and 4-93;

$$N_s = \frac{kN}{(k-1)s + 1} \quad (4-94)$$

Substituting Equation 4-94 in Equation 4-90 gives:

$$E_n = \frac{kN + (k-1)s + 1}{2[(k-1)s + 1]} \quad (4-95)$$

The expected value of the number of subject headings and file items examined to retrieve one file item in this type of file organization is Equation 4-88 plus Equation 4-95:

$$E = \frac{kN + (k-1)s + 1}{2[(k-1)s + 1]} + \left[\frac{k+1}{2}\right] \log_k [(k-1)s + 1] + \frac{1-k}{2} \quad (4-96)$$

It is now evident that when file items are related it may be possible to arrange each set of N_s items so that it can be searched logarithmically. In this case Equation 4-96 becomes:

$$E = \left[\frac{(k-1)N + s_L}{(k-1)N}\right] \log_k \left[(k-1) \frac{N}{s_L} + 1\right] - \frac{1}{k-1} + \left[\frac{k+1}{2}\right] \log_k [(k-1)s + 1] + \frac{1-k}{2} \quad (4-97)$$

Equation 4-97 is obtained from Equations 4-83, 4-88, and 4-89. Equation 4-93 was used to obtain the value of s_L .

4.4.1.3.3 Case III - This case is the same as Case I except that the tree is not indexed. That is, any node may be a satisfactory response to a request; but after selecting a node at a given level, it is necessary to examine the nodes at the next lower level connected to the selected node in order to ascertain which one is the next appropriate subheading.

In this case the maximum number of nodes examined at each level except the first is simply k . The number of nodes examined at the first level is 1. Therefore, the maximum number of nodes examined in any search is:

$$n = k(L - 1) + 1 \quad (4-98)$$

hence, from Equations 4-82 and 4-98:

$$n = k \log_k [(k - 1)N + 1] + (1 - k) \quad (4-99)$$

Therefore, the expected number of nodes examined is:

$$\begin{aligned} E &= \sum_{i=1}^n \frac{1}{n} \\ &= \frac{k}{2} \log_k [(k - 1)N + 1] + \frac{2 - k}{2} \end{aligned} \quad (4-100)$$

where n is determined from Equation 4-99.

4.4.1.3.4 Case IV - This case considers an indexed tree of subject headings rather than file items with the file items located under the lowest row of nodes or subject headings. The equally likely assumption is involved, as usual. Two variations can be considered. First, the file items are sequential and searched in order. Second, the file items are searched logarithmically; in this variation the items are

actually filed in a tree structure.

Since the subject headings in this case are not responses, the expected number of headings examined is fixed and equal to the number of levels L in the tree. Therefore, from Equation 4-87:

$$E_s = \log_k [(k-1)s + 1] \quad (4-101)$$

For a sequentially searched file, the expected number of items searched is obtained from Equation 4-95. Therefore, the expected number of subject headings and items searched is:

$$E = \frac{kN + (k-1)s + 1}{2[(k-1)s + 1]} + \log_k [(k-1)s + 1] \quad (4-102)$$

If the items are searched logarithmically, the expected number is obtained by taking N equal to N_s and then substituting Equation 4-94 in Equation 4-83. The resulting equation is:

$$E_n = \left[\frac{(k-1)(kN + s) + 1}{k(k-1)N} \right] \log_k \left[\frac{(k-1)(kN + s) + 1}{(k-1)s + 1} \right] - \frac{1}{k-1} \quad (4-103)$$

Therefore, the expected number of subject headings and items examined is Equation 4-101 plus Equation 4-103:

$$E = \log_k [(k-1)s + 1] + \left[\frac{(k-1)(kN + s) + 1}{k(k-1)N} \right] \log_k \left[\frac{(k-1)(kN + s) + 1}{(k-1)s + 1} \right] - \frac{1}{k-1} \quad (4-104)$$

4.4.1.4 Analysis and Comparison of the Expected Values - The major purpose of deriving expressions for the expected values of the

TABLE 2. SUMMARY OF FILE ORGANIZATIONS

SUBJECT HEADINGS	ITEMS	CASE	AVERAGE NUMBER OF HEADINGS AND ITEMS EXAMINED TO FIND ONE ITEM	EQUATION NUMBER
Linear	Sequential under each heading	Single Level	$E = \frac{s+1}{2} + \frac{N+s}{2s} = \frac{1}{2}(s + N/s + 2)$	4-75
None	Indexed tree	Case I	$E = \frac{1}{N} \sum_{j=1}^L j k^{j-1} - \left[\frac{(k-1)N+1}{(k-1)N} \right] \log_k [(k-1)N+1] - \frac{1}{k-1}$	4-83
Non-indexed tree	Sequential under last row of nodes	Case II-A	$E = \frac{kN + (k-1)s + 1}{2[(k-1)s + 1]} + \left[\frac{k+1}{2} \right] \log_k [(k-1)s + 1] + \frac{1-k}{2}$	4-96
Non-indexed tree	Indexed trees under last row of headings	Case II-B	$E = \left[\frac{(k-1)N + s_L}{(k-1)N} \right] \log_k \left[(k-1) \frac{N}{s_L} + 1 \right] + \left[\frac{k+1}{2} \right] \log_k [(k-1)s + 1] - \frac{(k^2 - 2k + 3)}{2(k-1)}$	4-97

TABLE 2 (Continued). SUMMARY OF FILE ORGANIZATIONS

SUBJECT HEADINGS	ITEMS	CASE	AVERAGE NUMBER OF HEADINGS AND ITEMS EXAMINED TO FIND ONE ITEM	EQUATION NUMBER
None	Non-indexed tree	Case III	$E = \sum_{i=1}^r \frac{1}{n} = \frac{k}{2} \log_k [(k-1)N + 1] + \frac{2-k}{2}$	4-100
Indexed tree	Sequential under last row of nodes	Case IV-A	$E = \frac{kN + (k-1)s + 1}{2[(k-1)s + 1]} + \log_k [(k-1)s + 1]$	4-102
Indexed tree	Indexed trees under last row of headings	Case IV-B	$E = \log_k [(k-1)s + 1] + \left[\frac{(k-1)(kN + s) + 1}{k(k-1)N} \right] \bullet \log_k \left[\frac{(k-1)(kN + s) + 1}{(k-1)s + 1} \right] - \frac{1}{k-1}$	4-104

number of headings and items examined in various file structures is that these values provide a convenient (if oversimplified) means of comparing the effectiveness of different file structures. These file organizations and their corresponding average values are summarized in Table 2.

For general purposes of comparison the equations identified in Table 2 can be rewritten in simpler form. The simplified versions are given below with their original numbers followed by "A". The subscript s stands for subject headings; N for file items.

$$E = \frac{1}{2} [s + N/s + 2] = \frac{s+1}{2} + \frac{N_s+1}{2} \quad (4-75A)$$

where N_s is obtained from Equation 4-73.

$$E = L_N - \frac{1}{k-1} \quad \left(N \geq \frac{100}{k-1} \right) \quad (4-83A)$$

where $L_N = n$ is obtained from Equation 4-82.

$$E = \left[\frac{(k+1)}{2} \right] (L_s - 1) + 1 + \frac{N_s+1}{2} \quad (4-96A)$$

where L_s is obtained from Equation 4-87; N_s , from Equation 4-94.

$$E = \left[\frac{k+1}{2} \right] (L_s - 1) + 1 + L_{N_s} - \frac{1}{k-1} \quad \left(N \geq \frac{100}{k-1} \right) \quad (4-97A)$$

where L_s and L_{N_s} are obtained from Equation 4-87; N_s , from Equation 4-94.

$$E = \frac{k}{2}(L_N - 1) + 1 \quad (4-100A)$$

where $L_N = n$ is obtained from Equation 4-82.

$$E = L_s + \frac{N_s+1}{2} \quad (4-102A)$$

where L_s is obtained from Equation 4-87; N_s , from Equation 4-94.

$$E = L_s + L_{N_s} - \frac{1}{k-1} \quad \left(N \geq \frac{100}{k-1} \right) \quad (4-104A)$$

where L_s and L_{N_s} are obtained from Equation 4-87; N_s , from Equation 4-94.

These equations can be analyzed in two major ways with respect to E . The first is to ascertain within a given equation whether there is a relationship between s and N that will minimize E for that type of file organization. The second is to compare the equations with each other to determine whether some file structures are always superior to others.

To carry out the first analysis it is sufficient to assume that s can take any positive real value and to differentiate each of the equations with respect to s , considering N as a constant, and checking to see if the resulting extremum is indeed a minimum. If there is such a relationship between s and N , it provides the proper number of subject headings s to minimize E for a file of N items with that type of organization.*

*In the following discussion the values of s , which optimize the expected number of headings and items examined, are obtained for several of the file organizations. This derivation is accomplished by differentiating the expression for E with respect to s to obtain the appropriate s as a function of N that minimizes E . Strictly speaking, such a procedure is not permissible because all the distributions considered are discrete. E is defined only for positive integral values of s and N . Nevertheless, the equations for E in all cases are continuous functions for the domains of k , s , and N that are of interest. Consequently, these differentiations can be carried out formally and the relative minima obtained. To obtain the integral values of s that minimize E , it is then necessary to substitute the two integers closest to the minimum s into the equation for E to ascertain which gives the smaller E . This integer is then used as the minimum, provided it is positive. Even this procedure would not

For example, taking the partial derivative of E with respect to s in Equation 4-75A and setting the result equal to zero yields:

$$s = \sqrt{N} \quad (4-105)$$

A check reveals that the appropriate conditions for a minimum are satisfied. That is, the value of s given in Equation 4-105 will always result in a minimum E for that N. Substituting Equation 4-105 in Equation 4-75A gives:

$$E_{\min} = 1 + \sqrt{N} \quad (4-106)$$

From Equations 4-73 and 4-105, the optimum value for N_s is:

$$N_s = \sqrt{N} \quad (4-107)$$

Equation 4-38A cannot be treated in this manner because it is a function of N only (and k). It is true, however, that as k increases, E decreases. Care must be taken in the interpretation of this result.

Application of the same method to Equation 4-96A yields:

$$s = \frac{1}{k-1} \left[\frac{kN}{(k+1)\log_k e} - 1 \right] \quad (4-108)$$

This value of s for any N will yield the minimum E in Equation 4-96A.

The value of E is:

$$E_{\min} = 3/2 + \left[\frac{k+1}{2} \right] \log_k \left[\frac{eN}{(k+1)\log_k e} \right] \quad (4-109)$$

be sufficient were it not for the fact that these functions, in the cases considered, have only one relative minimum, and, therefore, this relative minimum is also an absolute minimum. The ultimate justification for these unrigorous techniques is that they do provide the real minima and, therefore, have considerable utility.

Equation 4-97A has no relative minimum. However, the optimum value for s can be obtained by observation. By substituting Equations 4-87 and 4-94 in 4-97A and simplifying, the result obtained is:

$$E = \left[\frac{k-1}{2} \right] [\log_k [(k-1)s + 1] - 1] + \log_k [k(k-1)N + (k-1)s + 1] - \frac{1}{k-1} \quad (4-97B)$$

This equation is defined for $s \geq 1$. For this range of s , Equation 4-97B has a minimum at $s = 1$. This minimum gives for E :

$$E = 1 + L_N - \frac{1}{k-1}$$

The single subject heading is superfluous and can be eliminated. The minimum E becomes:

$$E_{\min} = L_N - \frac{1}{k-1} \quad (4-110)$$

Therefore, the optimum s for Equation 4-97A is zero, and the equation has been reduced to Equation 4-83A. Consequently, it is disadvantageous to superimpose a non-indexed tree of subject headings on an indexed tree of file items.

Equation 4-100A is a function of N and k only; again, as k increases, E decreases.

For Equation 4-102A the s that gives minimum E is:

$$s = \frac{1}{k-1} \left[\frac{kN}{2 \log_k e} - 1 \right] \quad (4-111)$$

The minimum E becomes:

$$E_{\min} = \frac{3}{2} + \log_k \left[\frac{eN}{2 \log_k e} \right] \quad (4-112)$$

Equation 4-104A has no relative minimum. However, the optimum value for s can be obtained as follows. By substituting Equations 4-87 and 4-84 in Equation 4-104A and simplifying, it becomes:

$$E = \log_k [k(k-1)N + (k-1)s + 1] - \frac{1}{k-1} \quad (4-104B)$$

This equation is defined for $s \geq 1$. Obviously, it has an absolute minimum at $s = 1$, which gives:

$$E = 1 + L_N - \frac{1}{k-1}$$

The single subject heading again is superfluous, and E becomes:

$$E_{\min} = L_n - \frac{1}{k-1} \quad (4-113)$$

Thus the optimum s for Equation 4-104A is zero, and this equation is also reduced to Equation 4-83A. In other words, wherever it is possible to construct an indexed tree of items, it is pointless to superimpose an indexed tree of subject headings upon it. It is also pointless to establish any other system of subject headings. One example, namely Equation 4-97A, has already been considered.

The second type of analysis compares one equation with another for an arbitrary but specified file size N and for a number of headings s ; the objective is to determine whether E is always less in one type of file organization than in another. Equations 4-97A and 4-104A have been shown to be superfluous and will not be considered.

The files with no subject headings, Equations 4-83A and 4-100A will be considered first. For a given N , Equation 4-83A will yield a lower average number of items searched than Equation 4-100A if:

$$L_N - \frac{1}{k-1} < \frac{k}{2}(L_N - 1) + 1$$

This inequality can be written:

$$(L_N - 1) - \frac{1}{k-1} < \frac{k}{2}(L_N - 1) \quad (4-114)$$

The inequality is clearly valid for $k \geq 2$. Consequently, the average number of items examined in searching an indexed tree of N items is always less than the average number examined in a non-indexed tree.

For the case where the number of headings in both trees is the same, Equations 4-96A and 4-102A can be compared in terms of:

$$\left[\frac{k+1}{2}\right] (L_s - 1) + 1 > L_s$$

or

$$\left[\frac{k+1}{2}\right] (L_s - 1) > L_s - 1 \quad (4-115)$$

This inequality is clearly valid for $k \geq 2$ and $L_s \geq 1$. Therefore, Equation 4-102A gives a smaller E than Equation 4-96A. It is clear, however, from Equations 4-108 and 4-111 that the optimum s 's for the two trees of Equations 4-102A and 4-96A are not identical. Nevertheless, it can be shown directly from Equations 4-109 and 4-112 that Equation 4-102A also yields a smaller E than Equation 4-96A when s is optimized in each case. This optimization would require:

$$\left. \begin{aligned} \log_k \left[\frac{eN}{2 \log_k e} \right] &< \log_k \left[\frac{eN}{(k+1) \log_k e} \right]^{(k+1)/2} \\ \text{or} \quad \frac{eN}{2 \log_k e} &< \left[\frac{eN}{(k+1) \log_k e} \right]^{(k+1)/2} \end{aligned} \right\} \quad (4-116)$$

This inequality is valid for:

$$N > \frac{1}{2^{2/(k-1)}} \frac{(k+1)^{(k+1)/(k-1)}}{e \log_e k} \quad (4-117)$$

This condition presents no restriction for a practical case. For example, Equation 4-117 requires $N \geq 4$ if $k = 2$; $N \geq 3$, if $k = 10$; $N \geq 6$, if $k = 100$.

For a given N and a given $s > 1$, Equation 4-102A always gives a lower value of E than Equation 4-75A. The conditions would require:

$$L_s < \frac{s+1}{2}$$

This inequality can be transformed by algebra to:

$$k^{-(s+1)/2} [(k-1)s + 1] < 1 \quad (4-118)$$

By differentiating the left member of Equation 4-118 with respect to k and setting it equal to zero, a value for k can be obtained to make it an extremum. This value is:

$$k = \frac{s+1}{s} \quad (4-119)$$

By examining the second derivative at this point, it is observed that Equation 4-119 maximizes the left member of Equation 4-118 when $s > 1$.

This maximum value is:

$$2 \left[\frac{s}{s+1} \right]^{(s+1)/2} \quad (4-120)$$

For $s > 1$, the Value 4-120 is always less than 1. Since the maximum value satisfies Equation 4-118, any other value, in particular any $k \geq 2$, will also satisfy it.

When s is optimized in each case, these two file structures can

be compared by Equations 4-106 and 4-112. Equation 4-102A will give a lower E than Equation 4-75A in the optimum case when:

$$\frac{3}{2} + \log_k \left[\frac{eN}{2 \log_k e} \right] < \sqrt{N} + 1$$

By algebraic transformations, this inequality can be written:

$$\frac{N \ln k}{k^{\sqrt{N-1}}} < \frac{2}{e} \quad (4-121)$$

When $k = 2$, this inequality is valid for $N \geq 27$; when $k = 4$, it is valid for $N \geq 4$; when $k \geq 6$, it holds for $N \geq 1$.

The optimum cases of Equations 4-96A and 4-75A can be compared by using Equations 4-106 and 4-109. Equation 4-96A will yield a smaller E when:

$$\frac{3}{2} + \left[\frac{k+1}{2} \right] \log_k \left[\frac{eN}{(k+1) \log_k e} \right] < \sqrt{N} + 1$$

that is, when:

$$\frac{N \ln k}{[(k+1)k]^{\left[\frac{(2\sqrt{N-1})}{(k+1)} \right]}} < \frac{1}{e} \quad (4-122)$$

Equation 4-122 is generally valid for larger files. For example, a simple calculation with $k = 10$ shows that Equation 4-122 is valid for N roughly greater than 115 and invalid for smaller N. Hence, the single level subject heading file results in a smaller average number of items searched in files with less than 115 items. This conclusion is shown clearly in Figure 1.

Figure 1 depicts the average number of headings and items examined for a wide range of file sizes. Only optimum values for s

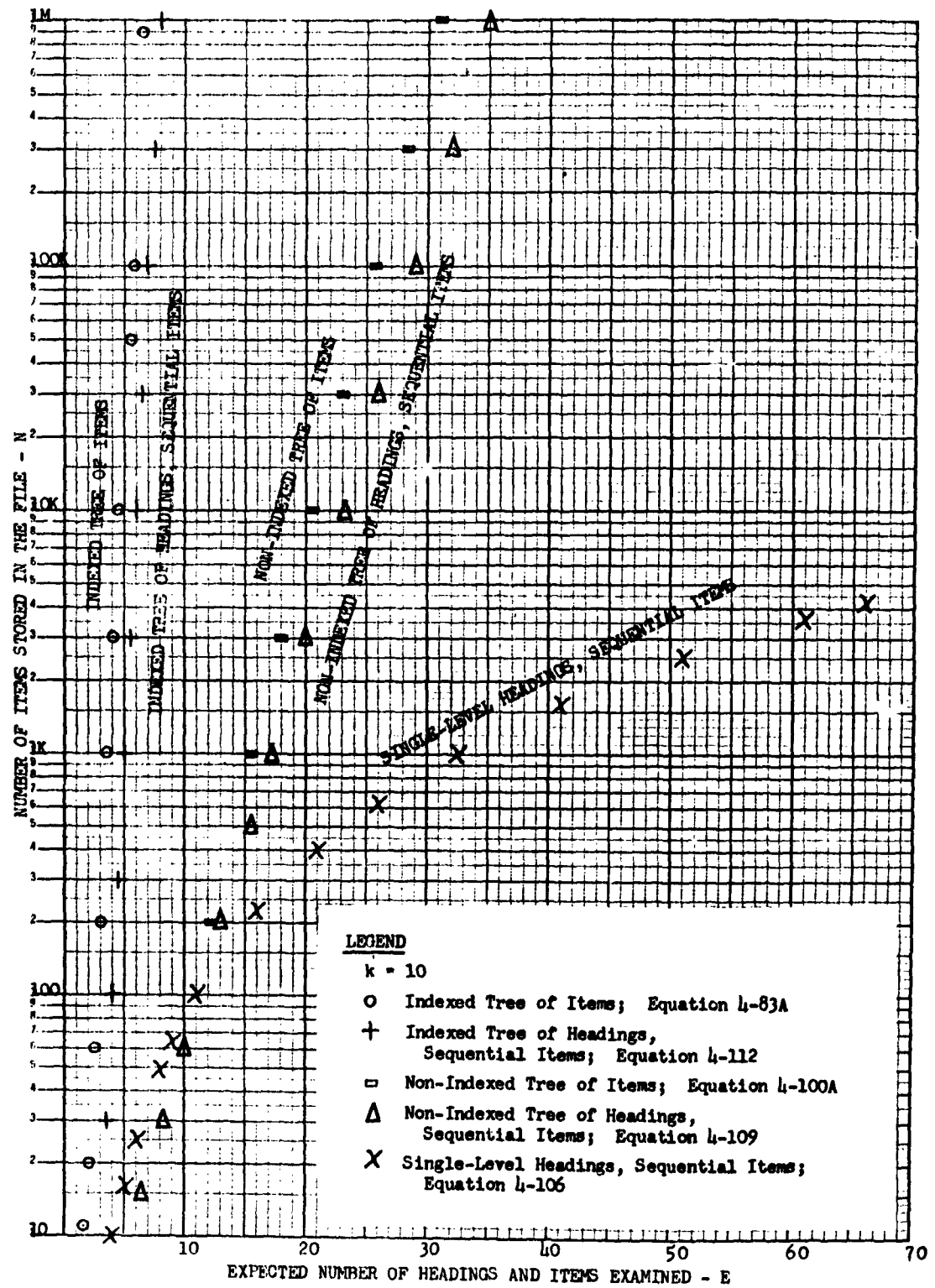


FIGURE 1. Average Number of Headings and Items Examined in a Search of Differently Organized Files

are shown. The figure indicates the superiority of indexed trees over non-indexed trees and of non-indexed trees over single-level subject headings, except for small files as indicated by Equation 4-122. However, the degree of superiority of the indexed trees is somewhat misleading. Although it is true that the average number of headings and items examined or searched for such trees is much smaller than for the other file structures, this fact does not imply much faster response times. By omitting consideration of the indexing function itself, the burden of search has in a sense merely been shifted elsewhere. Unless the indexing function is powerful, the search procedure in an indexed tree, particularly where k is large, may spend almost as much time examining indexes to determine the appropriate paths as would be involved in examining the headings themselves.

A singular feature of Figure 1 is that the indexed tree of items, Equation 4-83A, and the indexed tree of headings, Equation 4-102A, give similar values of E . The same is true for the non-indexed trees represented by Equations 4-100A and 4-96A. The explanation, however, is simple. Equations 4-108 and 4-111 require that the number of subject headings should be so large that essentially only a few items or even a single item are filed sequentially under each node of the last row. In other words, N_s is small. This fact can be seen from the values of N_s derived from Equations 4-94, 4-108, and 4-111, respectively. These values are:

$$N_s = (k + 1) \log_k e \quad (4-123A)$$

$$\left. \begin{array}{ll} N_s = 2 \log_k e & (k \leq 7) \\ N_s = 1 & (k > 7) \end{array} \right\} \quad (4-123B)$$

Consequently, almost all the searching is performed in the tree of headings where it is most economical. Hence, the close correspondence arises between trees of headings and between trees of items. Of course, in practice, it may frequently be impossible to achieve a meaningful breakdown of related headings to such a detailed level. Therefore, the optimum values of s , N_s , and E should be regarded as interesting idealizations. In practice, only integral values of s and N_s can be used.

In cases where the optimum curves plotted in Figure 1 are unrealistic because they restrict s too much, the equations developed in this and the previous section can be used to generate complete sets of design charts. From these charts the best file organization can be read, in terms of whatever value s must have to reflect the logical relationships and the nature of the subject matter to be classified.

In the interest of completeness, Figure 2 is included for reference. It relates the number of levels of nodes in a regular tree of order k to accommodate N items, one item per node. Figure 2 is obtained from Equation 4-82 or 4-87.

4.4.1.5 Variance From the Expected Values - The utility of the average or expected number of items and headings examined in different file structures depends upon the likelihood that the number of items and headings searched will generally be near the average value. An estimate

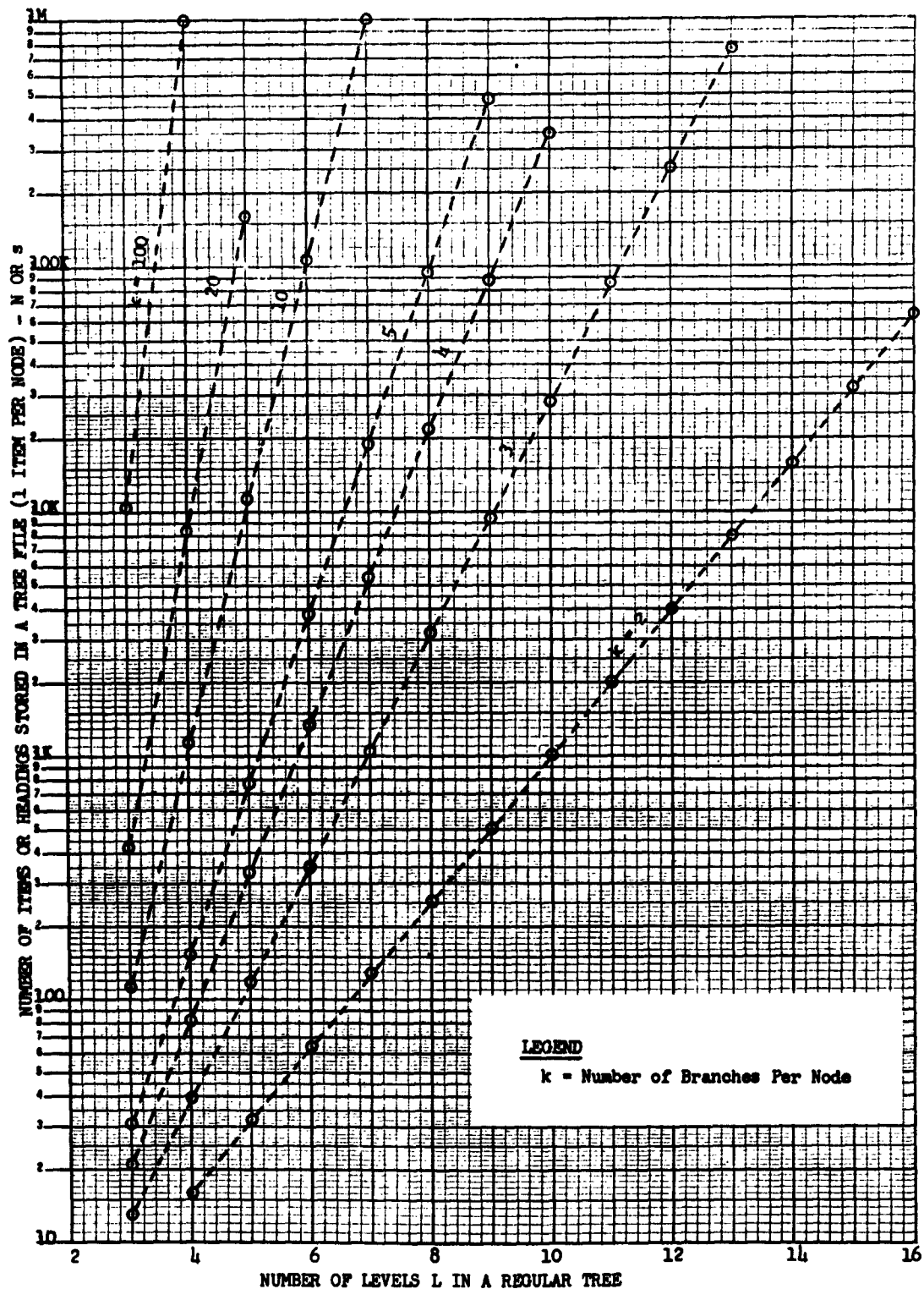


FIGURE 2. Number of Levels Required to Store N Items in a Regular Tree

of this likelihood is provided by the statistical variance of the number of items and headings searched from the average number. Expressions for the variance relative to Equations 4-74A, 4-83,* 4-96A, 4-100A, and 4-102A will be developed and analyzed.

Directly from the definition, the variance σ^2 of the single level subject heading file can be written:

$$\sigma^2 = \sum_{i=1}^s \frac{1}{s} [i - \frac{s+1}{2}]^2 + \sum_{i=1}^N \frac{1}{N} [i - \frac{N+s}{2s}]^2 \quad (4-124)$$

Carrying out the summations yields:

$$\sigma^2 = \frac{(s-1)(s+1)}{12} + \frac{(N/s)^2 - 1}{12} \quad (4-125)$$

$$\sigma_{(4-75A)}^2 = \frac{1}{12} [s^2 + (N/s)^2 - 2] \quad (4-126)$$

[Note: the subscript such as (4-75A) references the equation related to a given variance.]

By differentiating Equation 4-126 with respect to s , setting the result equal to zero, and checking the appropriate requirements, it can be shown that:

$$s = \sqrt{N} \quad (4-127)$$

gives the minimum variance. Thus the s that gives minimum E , Equations 4-105 and 4-106, also gives the minimum variance. This value is:

*In this case Equation 4-83 will be used instead of Equation 4-83A. Equation 4-83A is not sufficiently accurate to be used in computing the variances, because the variances are small. The computation is based upon differences between numbers that are approximately equal.

$$\sigma_{\min}^2 = \frac{N-1}{6} \quad (4-128)$$

For the indexed tree of items, the variance is:

$$\sigma^2 = \frac{1}{N} \sum_{j=1}^n k^{j-1} [j - E_{(4-83)}]^2 \quad (4-129)$$

where n is given by Equation 4-82. An elementary theorem of mathematical statistics states that Equation 4-129 is equal to:

$$\sigma^2 = \frac{1}{N} \sum_{j=1}^n j^2 k^{j-1} - E^2 \quad (4-130)$$

where E is the expected value obtained from Equation 4-83. The sum in Equation 4-130 can be evaluated by using some relationships among the derivatives of arithmetic and geometric series. Generating functions can also be employed directly and effectively, in this case, to obtain the variance. Using either of these methods, the following expression for the variance can be derived:

$$\begin{aligned} \sigma_{(4-83)}^2 = \frac{1}{k-1} \left[\frac{L_N^2}{N} - \frac{2L_N}{(k-1)N} - 2L_N + \frac{k+1}{k-1} \right] \\ + L_N^2 - E^2 \end{aligned} \quad (4-131)$$

where $n = L_N$ is obtained from Equation 4-82 and E from Equation 4-83. Equation 4-131 can be used to compute the variance for relatively small size files (moderately large N).

As N becomes arbitrarily large, however, Equation 4-131 approaches the following limiting value:

$$\sigma_{(4-83)}^2 = \frac{k}{(k-1)^2} \quad (4-132)$$

Equation 4-131 converges relatively rapidly to Equation 4-132. For example, when $k = 10$, the following errors in the variance are introduced by using Equation 4-132 rather than 4-131:

<u>N</u>	<u>Error in Equation 4-132</u>
10^3	1.11%
10^4	.70%
10^5	.05%

This point is primarily of academic interest, since the variances given by Equations 4-131 and 4-132 are insignificant. For $k \geq 3$, the variance given by Equation 4-131 is less than 1. It can be shown that the variance is a monotonically increasing function of N , and that Equation 4-132 is an upper limit for the variance.

Applying similar methods, the variances for the other file structures were derived. They are:

$$\sigma_{(4-96A)}^2 = \frac{(k+1)(k-1)}{12}(L_s - 1) + \frac{N_s^2 - 1}{12} \quad (4-133)$$

where L_s is obtained from Equation 4-87; N_s , from Equation 4-94.

$$\sigma_{(4-100A)}^2 = \frac{n^2 - 1}{12} \quad (4-134)$$

where n is obtained from Equation 4-99.

$$\sigma_{(4-102A)}^2 = \frac{N_s^2 - 1}{12} \quad (4-135)$$

where N_s is obtained from Equation 4-94.

The variances of Equations 4-96A and 4-102A can now be derived

for optimum s . From Equations 4-87 and 4-108:

$$L_{s_{opt}} = \log_k \left[\frac{kN}{(k+1)\log_k e} \right] \quad (4-136)$$

$$= 1 + \log_k \left[\frac{N}{(k+1)\log_k e} \right] \quad (4-137)$$

Substituting Equations 4-123A and 4-137 into Equation 4-133 yields:

$$\begin{aligned} \sigma_{(4-96A)_{opt}}^2 &= \frac{1}{12} \left\{ k^2 - 1 \right\} \log_k \left[\frac{N}{(k+1)\log_k e} \right] \\ &\quad + (k+1)^2 (\log_k e)^2 - 1 \end{aligned} \quad (4-138)$$

In the case of Equation 4-102A, substituting Equation 4-123B into Equation 4-135 gives:

$$\sigma_{(4-102A)_{opt}}^2 = \frac{4(\log_k e)^2 - 1}{12} \quad (4-139)$$

Whenever the optimum N_s given by Equation 4-123B is less than 1, N_s is taken as 1 and the variance given by Equation 4-139 is zero. The reason is, of course, that in this case there is a unique indexed procedure to locate any item in a fixed number of steps.

The standard deviations from the expected values are shown in Figure 3. In other words, Figure 3 is a graph of $\sigma_{(4-75A)_{opt}}$, $\sigma_{(4-83)}$, $\sigma_{(4-100A)}$, and $\sigma_{(4-96A)_{opt}}$ obtained by taking the positive square root of Equations 4-128, 4-131, 4-134, and 4-138, respectively. The graph was plotted for $k = 10$. For this value of k , the standard deviation of the indexed tree of headings with sequential items is zero for the reason given after Equation 4-139. Consequently, this standard deviation has not been included from the graph. As Figure 3 indicates, the standard

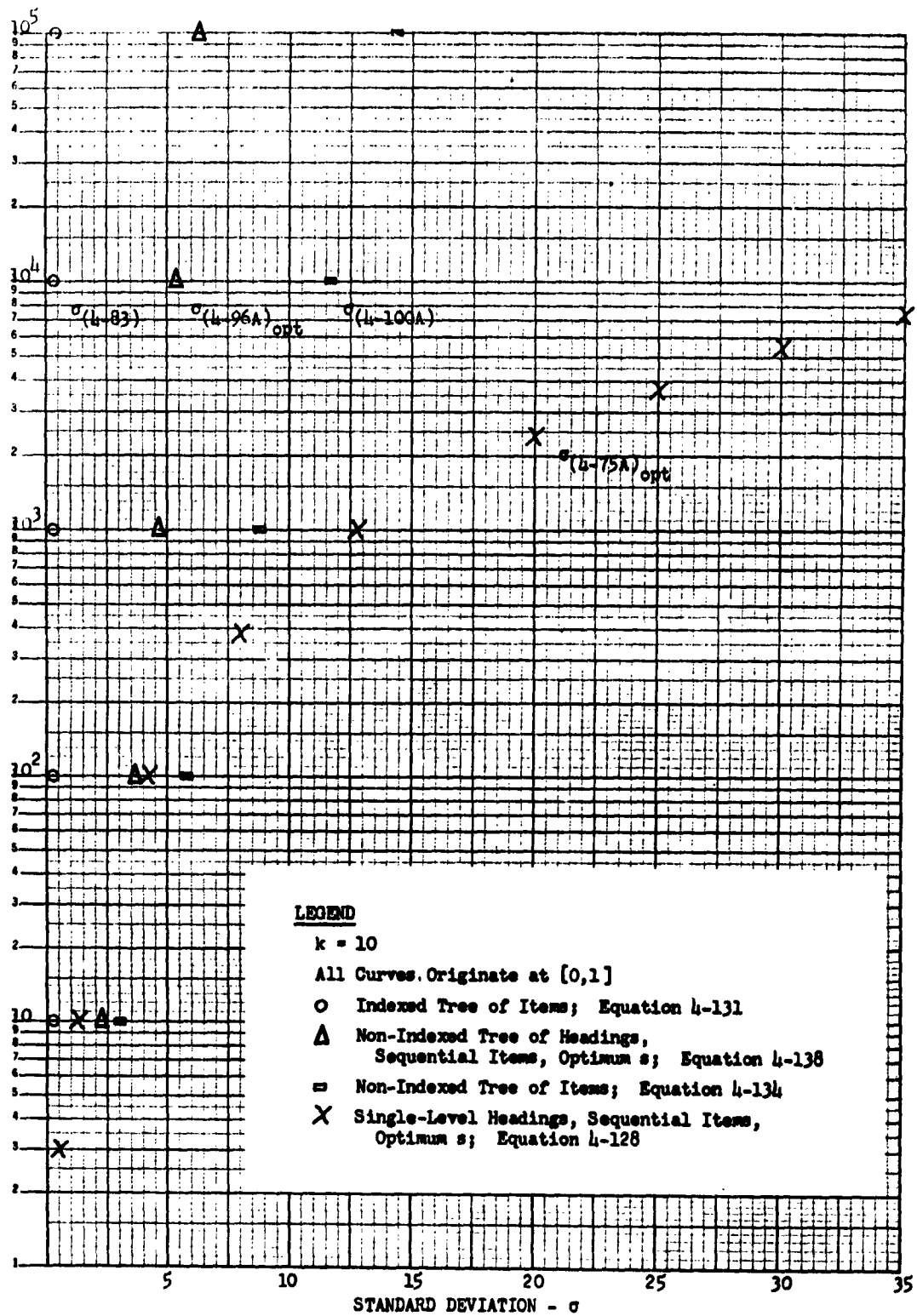


FIGURE 3. Standard Deviation From Average Number of Headings and Items Examined in a Search

deviation of the indexed tree of items, Equation 4-131, is also negligible. Hence, the expected value is a good indicator of the actual number of headings and items examined in a single search of an indexed tree. The standard deviation for the non-indexed tree of headings, Equation 4-138, is somewhat larger; for the non-indexed tree of items, Equation 4-134, it is still larger. For reasonably large files, the largest deviation is the single level subject heading file, Equation 4-128. Consequently, the expected number of headings and items examined is not a good indicator of what will occur in any given search of a single level file. This point is verified by anyone's experience with this kind of file.

Figure 4 compares the cumulative probability distributions for three types of files. It indicates rather clearly the wide variation in n among the file types (with a fixed file size) for any given probability that the number of headings and items searched will be not greater than n in any single search. For example, in a file of 111,111 items the probability is .5 that fewer than 7 items will be examined in an indexed tree; fewer than 25 in a non-indexed tree; but fewer than 335 in a sequential, single level heading file.

4.4.1.6 Generalized Expressions for Expected Values - The purpose of this section is to present generalized expressions for the expected number of headings and items searched, when two previous assumptions are removed. These assumptions are:

- (a) Each subject heading or item is equally likely to be the one sought.
- (b) The same number of items is filed under each heading.

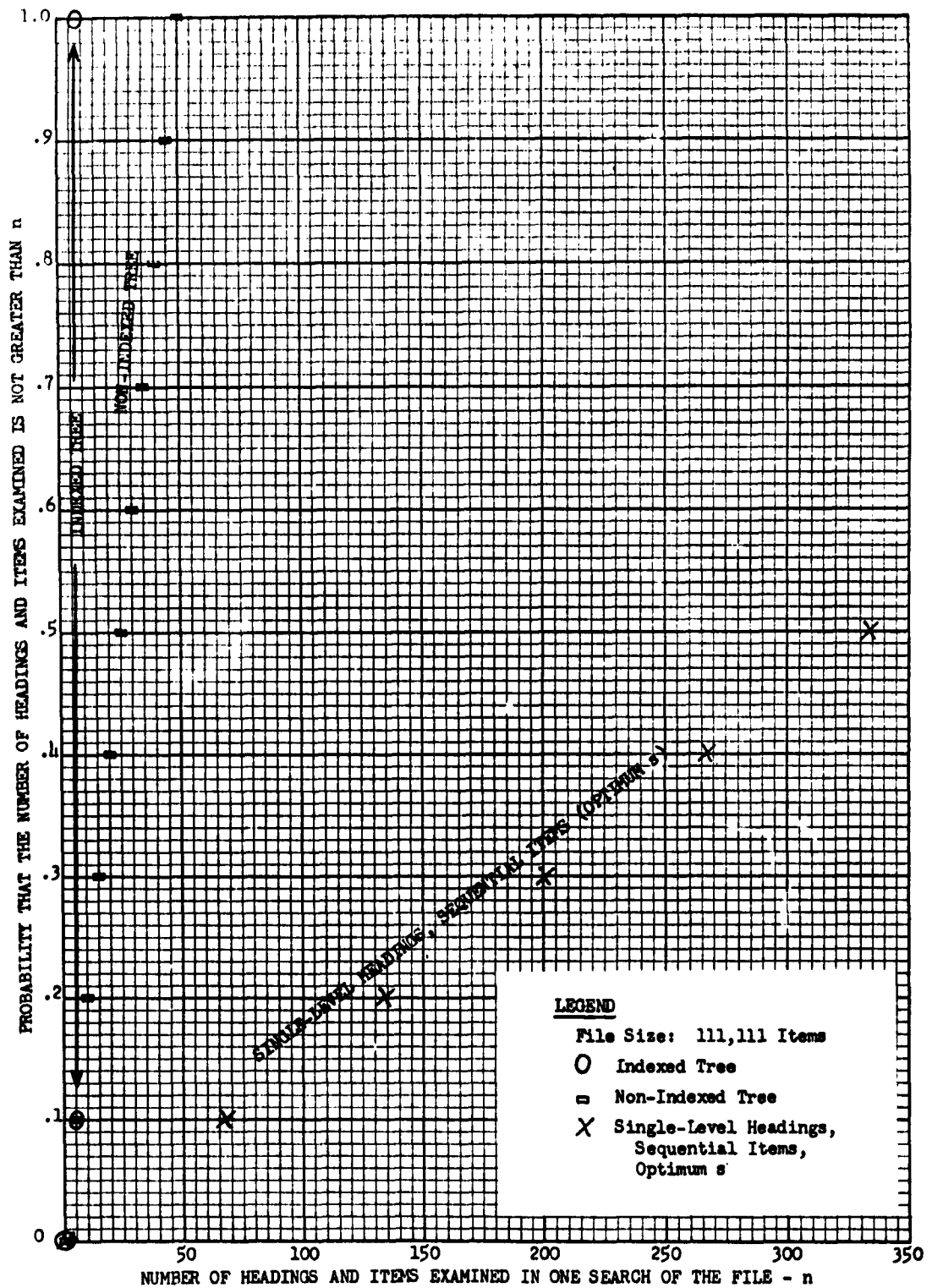


FIGURE 4. Cumulative Probability Distributions for a Search of Differently Organized Files

For example, if information is available on anticipated or past activity of the file items--and if this information indicates the likelihood of a given heading or item being requested--then the expected number of headings and items searched can be obtained in terms of the available data that approximate the probability distribution of file activity. Generally, the more specialized the contents of a file, the better known and more stable will be its activity. When the activity of the file is known and it is relatively stable, it is clearly advantageous to organize the file so that the items that have the greatest likelihood of being requested are the most accessible. For obvious reasons such a file is called activity organized. It is the intent of this section to provide a general background for the investigation of activity organized files in terms similar to those appearing in previous sections. For the sake of simplicity, expressions for expected values will be presented for only two of the file organizations. These expressions will provide a starting point for the analysis of activity organized files. In each case, $p(i)$ indicates the probability that the i^{th} item or heading is the answer to a request.

The single level subject headings with sequential items, Equation 4-75, generalizes to:

$$E = \sum_{i=1}^s i p_s(i) + \sum_{i=1}^s \left[\sum_{j=1}^{n_i} j p_i(j) \right] p_s(i) \quad (4-140)$$

where s = the number of subject headings in the file.

n_i = the number of items under heading i .

$p_s(i)$ = the probability that the answer to a request is under heading i .

$p(j)$ = the probability that item j is the answer to a request.

$p_1(j)$ = the probability that item j will be requested, given that it is filed under heading 1.

This last probability is obtained from:

$$p(j) = p_s(1) \cdot p_1(j) \quad (4-141)$$

The expected value for the indexed tree of items, Equation 4-83, generalizes to:

$$E = \sum_{j=1}^n j p(j) \quad (4-142)$$

where $p(j)$ is the probability of finding the answer on the j^{th} cut; it is given by:

$$p(j) = \sum_{i=1}^{k^{j-1}} p_j(i) \quad (4-143)$$

where $p_j(i)$ is the probability that the i^{th} node on level j is the requested item. Values for n are obtained from Equation 4-82.

4.4.1.7 Summary - Conclusions have been developed and presented throughout this section and will be summarized only briefly. These conclusions are valid only for files where every heading and item is equally likely to be required for a response.

- (a) In terms of expected values, indexed trees give a lower average number of headings and items examined than non-indexed trees. Non-indexed trees give lower values than single level subject headings, except for small files. The break-even points can be determined precisely from the equations in Section 4.4.1.4.
- (b) Whenever a file of items can be indexed or ordered into a tree

structure, it is disadvantageous, in terms of expected values, to superimpose any heading structure on the items.

- (c) For trees and single level subject heading files relationships between the number of headings and the number of items in the file minimize the expected number of headings and items that will be examined in a file search.
- (d) The standard deviation from the average number of headings and items examined for indexed trees is small. Consequently, these average numbers are excellent indicators of the number of headings and items likely to be examined in a single search. The deviations for non-indexed trees are somewhat larger, so expected values are of less utility. Finally, the deviation from the expected values of the file with single level headings and sequential items is so large that the average values are poor indicators of the number of headings and items examined in any single search.

This study can be extended in any one of several directions. The choice should be made on the basis of how well the work can be integrated with other research tasks in this project. The utility anticipated from these extensions should also be considered. Some general areas for possible further investigation are:

- (a) Extend the study to obtain required search times--i.e., mean recurrent events, Reference [5]--after taking into account the time required for indexing and other processing functions

necessary for retrieval.

- (b) Analyze other file organizations. Activity organized files should be investigated for several widely differing distributions to ascertain their advantage in terms of quantitative statistics. Files consisting of many related or unrelated trees and non-regular trees should also be considered.
- (c) Consider other models of file organization than tree structures--e.g., Markov chains--for the representation of the relationship between file organization and search.

4.5 INTEGRATIVE CAPABILITIES

The work on non-Boolean retrieval and on the comparative analysis of file organizations both have implications for integrative system models. To date, however, no explicit attempt at the formulation of such a model has been attempted. Preliminary theoretical speculation continually takes place. One area in which there has been an attempt to document such speculation concerns the relationship between frequency and indexing.

4.5.1 General Theoretical Considerations with Special Reference to the Relationship Between Frequency and Indexing - In a collection of n items, there is only a finite number of subcollections of items that are theoretically possible responses in item retrieval systems. The number is 2^n if zero items are considered a subcollection. In practice, not all 2^n answers are equally likely to be searched for by a user. Intuition suggests that this disparity is an essential criterion for the effective design of a query or descriptor language.

There are several possible approaches to specifying which of these 2^n subcollections is being referenced. In one sense the simplest means of specification is to assign a name or descriptor to each of the n items in the collection. In the case when all 2^n subcollections are requested equally often and when the questioner knows the name of each item he is interested in, this method produces an adequate system. If, however, some subcollections are considerably more popular than others, then an obvious improvement in coding efficiency would result from giving popular collections special category names.

There are, however, other considerations than information theoretic measures of coding efficiency that are relevant to the selection of a descriptor language. Asking for all the items in a subcollection by name is possible only when the names of all the documents in the subcollection that are of interest are known. Under these circumstances the general problem of information retrieval becomes a special case, and only considerations of coding efficiency and, perhaps, user compatibility are relevant criteria for descriptor language design.

In an ordinary library search the questioner does not know the names of the items he needs. He wants the system to supply a subcollection of items that will provide information relevant to his query after he reads them. The system must go from his query or a transformation of his query to an appropriate subcollection of items, even though the user does not yet know in advance what is in this subcollection.

How can the system do this? One approach is to ask, perhaps implicitly,

questions in advance and to search, again implicitly, the entire collection to find the items that contain information relevant to each question. The system would then have the stored answer available whenever the same question arose. In a sizable collection it is not feasible to ask all questions in advance. There are two reasons: first, there are a large number of ways of asking essentially the same question; another way of putting this point is that the same answer subcollection would satisfy many possible question variations. Second, there are too many possible answers--specifically, 2^n --in any sizable system.

Each of these difficulties requires a different approach. The approach to the former involves standardization; that is, the possible ways of asking essentially the same question must be restricted. This solution is essentially a language problem. The approach to the latter difficulties involves exclusion of less probable questions and their resultant answers from advance treatment. This solution is essentially a system design and organization problem.

How is explicit or implicit advance treatment of questions possible? One method would be to have all documents in the library unordered, except perhaps by author and title for those searches in which the querier already knows which documents he wants. Anyone wishing to use the library could then be asked to submit both a copy of his question and a list of the documents he found relevant after making his search of the library. This information could then be stored for occasions when the same or similar questions are asked.

Of course, this scheme is impractical. Listing some of its inherent difficulties may lead to an understanding of the requirements of an ideal descriptor-query language.

- (a) There is no assurance that any initial questioner will do a good or thorough job in searching all the documents in the library.
- (b) Even if the initial questioner has done a perfect job at the time he searched the library, there would be a lack of information about the relevance of new accessions to the question. Of course, new accessions could be re-searched by subsequent questioners in order to keep the answer list up to date.
- (c) Many questions will recur imprecisely; and even if the statement of the question is identical, different users are likely to have different meanings or intentions that would influence which documents they considered appropriate for the answer list. Thus, even if there is a perfect and up-to-date search performed by the initial questioner, it is not likely to be perfect for a subsequent questioner.
- (d) Such a system would impose an unacceptable search burden not only upon initial questioners but also upon subsequent questioners, if there are a substantial number of new acquisitions. Furthermore, the askers of somewhat unusual questions would always tend to be in the role of initial questioners, regardless of how long the system has been in operation. Their extensive search efforts would rarely be applied by subsequent users.

The technique currently used by most libraries, in order to deal with these objections, is implicitly to select a range of questions to be pre-answered and then to assess the relevance of each accession--i.e., index it--to all these questions as it is entered into the library file. To the extent that a document's relevance to many questions can be assessed nearly simultaneously, this technique has obvious advantages over repeatedly scanning each document for each question in some sequence of questions.

The approach of classifying each accession for all questions will deal completely only with difficulties (a) and (b). Difficulties (c) and (d) will be resolved only to the extent that the question list, against which each document is implicitly being checked, is sufficiently extensive and to the extent that the meaning of these implicit questions is sufficiently clear to the system users.

It is likely that none of the difficulties will ever be resolved completely. Even a user searching on the basis of his own question is likely to introduce inadvertent errors of both inclusion and exclusion on the answer list if he is scanning a large file collection. Similar errors will occur when a librarian classifies a book. But additional errors will result from the fact that the meaning of the implicit questions reflected by the classification varies from person to person.

These errors, while often significant, are not as basic a problem as the limitation on possible questions that can be answered. These limitations are a necessary concomitant of indexing a large collection. As has already been suggested, there are two kinds of limitations:

- (a) Basic limitations on the retrieval of all 2^n answers. In general, no indexing scheme for a sizable collection is sufficiently articulated to allow retrieval of all possible answers without knowing the names of individual documents.
- (b) Secondary limitations on the acceptability, or communicability, of a specific question formulation that does in fact correspond to one of the accessible answers.

The latter limitation does not necessarily imply any change in the logical organization of the indexing or query-descriptor language. The problem is one of using appropriate names or labels for the index terms or combinations of index terms that correspond to those of the 2^n answers that the system is capable of generating. Of course, the problem is not one that can be solved merely by the judicious selection of terms. It is necessary that the questioner and the library system use these terms in essentially the same sense. Furthermore, it is necessary that alternate descriptions of the same answer or question be interconvertible, either by the library system or by the user. To date, the only methods of dealing with this problem have been to provide the user with a dictionary-type description of the index terms, an over-view of the relationship among the terms used by the system, and/or a thesaurus type of referral ("see" and "see also") to related terms.

The problem of converting synonymous descriptions probably cannot be approached by considering the relative frequency of subcollection questions. Of course, the more popular a subcollection, the more valuable it might be to be able to deal with alternate ways of describing it. The problem of unaskable questions, however, can only be approached fruitfully from this point of view. If the system is to be insufficiently articulated for the

retrieval of all 2^n possible answer collections, it seems that the criteria (other than random exclusion based upon cost considerations) for deciding which subcollections are to be retrievable should ultimately be based upon the frequency of user demand. Only those questions that will rarely or never be asked should in principle be unanswerable--without searching the entire collection--because of limitations in the query language and the accompanying file structures and search procedures.

This conclusion suggests that a second consideration, besides the relative frequency of user demand for various possible answers, may be important. This consideration is the absolute level of demand for a possible answer subcollection. The absolute level of demand is readily calculated from estimates of relative demand and the total number of questions asked. An estimate for the number of questions may be the length of time for which the collection of items will be used multiplied by levels of use such as questions per day during this interval. As absolute use of the system as a whole increases, more articulate indexing becomes necessary to include the relatively less frequently asked questions, which now are asked a significant number of times in the system's lifetime.

Answer subcollections should not merely be regarded as accessible or inaccessible with a given query capability. Even if a subcollection is not immediately accessible, there are degrees of desirability that can be discriminated with in regard to its inaccessibility. Thus a desired answer subcollection may not be directly accessible per se, yet it may be wholly embedded in another subcollection that is accessible and that contains few additional items. Clearly, there is no great deficiency in query capability

under such circumstances so long as the user can identify and ask for the appropriate inexact subcollection. If, however, the items in a desired inaccessible subcollection are widely scattered--that is, the items cannot be obtained without searching a number of accessible subcollections--the situation is quite different. This difficulty is likely to be further complicated by the inherent unavailability of information about which accessible subcollections contain the items the user needs. Under such circumstances the user may be reduced to searching the entire collection, or unacceptably large parts of it, in order to obtain the needed information. It might be fruitful to develop rigorous measures of degree of inaccessibility based upon minimal and/or maximal false drops and/or misses.

Such a measure of accessibility could be used to evaluate the goodness of any descriptor scheme for any item collection. More precisely, it could be used to measure the average (in)accessibility for the power set of items, the set of 2^n possible answers, for a given descriptor scheme. When combined with information about relative frequencies of the members of the possible answer set, such a measure can provide information about the average accessibility of items per request. The main purpose of a general theory of information retrieval is to provide an analytical framework in which this quantity, the average accessibility per request, can be optimized, given a context of relevant system parameters.

Some of the relevant parameters that such a model should ultimately encompass are:

- (a) Number of items in the system.

- (b) Number of descriptors.
- (c) Articulation of descriptor scheme.
- (d) Cost per descriptor assignment.
- (e) Cost per false drop.
- (f) Cost per miss.
- (g) Cost per search unit.
- (h) Cost per file unit.
- (i) Number of queries.

The development of models for estimating the cost parameters is an important problem on which further work is necessary--see Reference [6]. Such quantities can, however, be treated parametrically in a general information retrieval model, and valuable insight into the design of optimal descriptor schemes may thus be obtained.

It may be objected that the basic datum of such a model--viz., estimates of relative frequency of reference to members of the power set of items--is virtually impossible to obtain in detail. It is undoubtedly both impossible to get completely accurate estimates and impractical to get even inaccurate estimates for each member of the power set of a substantial number of items. This difficulty does not, however, preclude parametric treatment of the distribution of relative frequency of reference among the members of the power set.

It is possible to estimate intuitively some of the consequences of the relative frequency of reference distribution in the answer set. If all answers are equally probable, there would seem to be no basis for choosing among which answers should be accessible. Under these circumstances a

uniterm type of descriptor system, in which there are no hierarchical organizations, might be most efficient. If, on the other hand, it is clear that many members of the answer set are answers in principle only and that such a collection of items would rarely be called for, then a hierarchical organization of the index may be appropriate. Similarly, when the cost of false drops is relatively high then, for a given number of descriptors, the average number of documents referenced per descriptor should be relatively small. If the cost of misses is emphasized, however, then the average number of documents per descriptor should be relatively large.

A rigorously formulated model would test and add quantitative depth to such intuitive conclusions and would probably generate other unforeseen, but perhaps more significant, relationships.

4.6 REFERENCES

- [1] Luhn, H. P., "The Automatic Creation of Literature Abstracts," IBM Journal of Research and Development, 2:159-165, April 1958.
- [2] Giuliano, Vincent E., Studies for the Design of an English Command and Control Language System, Arthur D. Little, Inc., Cambridge, Massachusetts, June 1962.
- [3] Newell, A., and Simon, H. A., The Simulation of Human Thought (AD 235-801); RAND Corporation, RM-2506, December 28, 1959.
- [4] Maron, M. E., "Automatic Indexing: An Experimental Inquiry," Journal of ACM, 8:404, 1961.
- [5] Feller, W., An Introduction to Probability Theory and Its Applications, Volume I, John Wiley and Sons, Inc.
- [6] Hayes, G., Mathematical Models for Information System Design and a Calculus of Operations, Final Report, Advanced Information Systems Co., Air Force Contract AF 30(602)-2111, 1961.

-----, Report on the Organization of Large Files with
Self-Organizing Capability, Advanced Information Systems
Co., National Science Foundation Contract C 162, 1961.

5. CONCLUSIONS

All areas of capability have been extended by analytical study of aspects of the information retrieval problem that required fuller definition and articulation. Input capabilities have been specifically dealt with from the viewpoints of using word frequency as an indicator in automatic indexing and of using a non-Boolean retrieval scheme.

Query capabilities were analyzed for the purposes of automatic extracting and redundancy control. Organization and search schemes were specified and their implications compared. A preliminary consideration of the relation of frequency to the assessment of indexing and thus to a model for system integration was presented.

Most of these contributions are still essentially in the analytical and research stage. The only area that could currently proceed to experimental implementation is the work on automatic extracting. Because of the magnitude of such a task and its subordinate position in the project as a whole, it is recommended that this work be continued as a separate project.

6. PLANS FOR NEXT QUARTER

Activities during the next quarter will proceed with the over-all goal of developing a theory of information retrieval for use as a tool in the design of information retrieval systems. This work will proceed within the specific task framework described in this report. The general emphasis will continue to be analytical with the primary purpose of developing methods to evaluate the relationship among significant system parameters. Toward these ends work on literature accession and review will continue to be a significant feature of the next quarter's activities.

Under input capability work will continue on problems of automatic classification with a view to generalizing to the case of non-exclusive classes. Planned extensions of this work include work on the optimal definition and location of class boundaries and on the evaluation of the adequacy of prediction and classification schemes. Work on non-Boolean retrieval will be continued. As already indicated, this work has implications for more general areas of capability.

Under query capabilities no extension of presently reported work on automatic extracting is planned. Further extensions on redundancy are, however, being considered. The formulation of a general theory of descriptor languages based upon frequency and accessibility will have important implications for improving query capabilities.

Under processing capabilities it is planned to focus on the problem of associative techniques--an area that has been relatively untouched

during this quarter. Some specific possibilities for extension on organization and search have been enumerated. A special intensive review of the applicability of the multi-list scheme of Prywes and Gray is also planned for this area.

Under integrative capabilities it is planned to attempt more rigorous formulations of the kind of system model alluded to in Section 4.5.1. Further documentation in the area of general theoretical considerations may also be expected.

7. IDENTIFICATION OF PERSONNEL

7.1 PERSONNEL ASSIGNMENTS

The following personnel were assigned to the project during the period covered by this report:

<u>Name</u>	<u>Title</u>	<u>Man-Hours</u>
Jacques Harlow	Manager	50
Quentin A. Darmstadt	Research Specialist	350
George Greenberg	Senior Specialist	400
Alfred Trachtenberg	Senior Program Analyst	450
Alexander Szejman	Senior Specialist	150

7.2 BACKGROUND OF PERSONNEL

The backgrounds of personnel originally assigned to the project were described in the First Quarterly Report. One new person was assigned to the project this quarter. A description of his background follows.

7.2.1 Alexander Szejman - BS, Physics, City College of New York, 1956; MA, Mathematical Economics, New York University, 1962; Graduate work in Physics, New York University. Activities involve mathematical analyses of adaptive and learning information systems. Previous experience includes mathematical analysis of diverse engineering problems and computer simulation.

DISTRIBUTION LIST

<u>Recipient</u>	<u>Copies</u>
OASD (R&E) Rm 3E1065 Attention: Technical Library The Pentagon Washington 25, D. C.	1
Chief of Research and Development OCS, Department of the Army Washington 25, D. C.	1
Commanding General U. S. Army Materiel Command Attention: R&D Directorate, Res Div, Elect Br. Washington 25, D. C.	1
Commanding General U. S. Army Electronics Command Attention: AMSEL-AD Fort Monmouth, New Jersey	3
Commander, Armed Services Technical Information Agency Attention: TIPOR Arlington Hall Station Arlington 12, Virginia	(Reports) 10
Commanding General USA Combat Developments Command Attention: CDCMR-E Fort Belvoir, Virginia	1
Commanding Officer USA Communication and Electronics Combat Development Agency Fort Huachuca, Arizona	1
Commanding General U. S. Army Electronics Research and Development Activity Attention: Technical Library Fort Huachuca, Arizona	1

<u>Recipient</u>	<u>Copies</u>
Chief, U. S. Army Security Agency Arlington Hall Station Arlington 12, Virginia	2
Deputy President U. S. Army Security Agency Board Arlington Hall Station Arlington 12, Virginia	1
Director, U. S. Naval Research Laboratory Attention: Code 2027 Washington 25, D. C.	1
Commanding Officer and Director U. S. Navy Electronics Laboratory San Diego 52, California	1
Aeronautical Systems Division Attention: ASAPRL Wright-Patterson Air Force Base, Ohio	1
Air Force Cambridge Research Laboratories Attention: CRZC L. G. Hanscom Field Bedford, Massachusetts	1
Air Force Cambridge Research Laboratories Attention: GRXL-R L. G. Hanscom Field Bedford, Massachusetts	1
Headquarters, Electronic Systems Division Attention: ESAT L. G. Hanscom Field Bedford, Massachusetts	1
Rome Air Development Center Attention: RAALD Griffiss Air Force Base, New York	1

<u>Recipient</u>	<u>Copies</u>
AFSC Scientific/Technical Liaison Office U. S. Naval Air Development Center Johnsville, Pennsylvania	1
Commanding Officer U. S. Army Electronics Materiel Support Agency Attention: SELMS-ADJ Fort Monmouth, New Jersey	1
Director, Fort Monmouth, Office USA Communication and Electronics Combat Development Agency Fort Monmouth, New Jersey	1
Corps of Engineers Liaison Office U. S. Army Electronics Research & Development Laboratory Fort Monmouth, New Jersey	1
Marine Corps Liaison Office U. S. Army Electronics Research & Development Laboratory Fort Monmouth, New Jersey	1
AFSC Scientific/Technical Liaison Office U. S. Army Electronics Research & Development Laboratory Fort Monmouth, New Jersey	1
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Logistics Division Fort Monmouth, New Jersey Attention: Anthony V. Campi	9
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Director of Research/Engineering Fort Monmouth, New Jersey	2
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Technical Documents Center Fort Monmouth, New Jersey	2

<u>Recipient</u>	<u>Copies</u>
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: SELRA/NPE Fort Monmouth, New Jersey	2
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Technical Information Division Fort Monmouth, New Jersey	3
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Exploratory Research Dr. Reilly Fort Monmouth, New Jersey	2
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Engineering Sciences Department Mr. Hennessy Fort Monmouth, New Jersey	2
Commanding Officer U. S. Army Electronics Research & Development Laboratory Attention: Exploratory Research Jack Benson Fort Monmouth, New Jersey	2
Headquarters, Aeronautical Systems Division Air Force Systems Command, USAF Attention: ASRCM-1 (Mr. Thompson) Wright-Patterson Air Force Base, Ohio	1